Cy 1

AD A056 966

AVRADCOM Report No. 78-29

Production Engineering Measures Program
Manufacturing Methods and Technology

COMPUTER-AIDED DESIGN AND MANUFACTURING FOR EXTRUSION OF ALUMINUM, TITANIUM AND STEEL STRUCTURAL PARTS

PHASE II - APPLICATION TO PRACTICAL EXTRUSIONS

VIJAY NAGPAL, CARL F. BILLHARDT AND TAYLAN ALTAN
BATTELLE, Columbus Laboratories
505 King Avenue
Columbus, Ohio 43201

January 1978

AMMRC TR 78-26

Final Report Contract Number DAAG46-75-C-0054

VOLUME II

Approved for public release; distribution unlimited.

Prepared for
U.S. ARMY AVIATION RESEARCH AND DEVELOPMENT COMMAND
St. Louis, Missouri 63166

ARMY MATERIALS AND MECHANICS RESEARCH CENTER
Watertown, Massachusetts 02172

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AVRADCOM TR 78-29	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Computer-Aided Design and Manufacturing for Extrusion of Aluminum, Titanium and Steel Structural Parts - Phase II - Application of CAD/CAM to Practical Extrusions		5. TYPE OF REPORT & PERIOD COVERED Final 7/19/76 - 10/18/77 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Vijay Nagpal, Carl F. Billhardt and T. Altan		8. CONTRACT OR GRANT NUMBER(s) DAAG46-75-C-0054
9. PERFORMING ORGANIZATION NAME AND ADDRESS Battelle's Columbus Laboratories 505 King Avenue Columbus, Ohio 43201		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS AMCMS Code:1497.94.5.S8154 (xR5)
11. CONTROLLING OFFICE NAME AND ADDRESS U. S. Army Aviation Research and Development Command, ATTN: EXT, P. O. Box 209 St. Louis, MO 63166		12. REPORT DATE January 1978 13. NUMBER OF PAGES
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) U. S. Army Materials and Mechanics Research Center Watertown, Massachusetts 02172		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

AMMRC TR 78-26

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Computer-Aided Design
CAD/CAM
Extrusion
Die Design
Aluminum Alloys
Titanium Alloys
Steel

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The overall objective of this program was to develop practical computer-aided design and manufacturing (CAD/CAM) techniques for extrusion of aluminum alloys, steels, and titanium alloys.

This program was conducted in two phases. This report covers the Phase-II work, which was completed by performing the following major tasks: (a) assemble geometric modules to represent practical exturusion shapes, (b) apply the CAD/CAM method to a streamlined die, (c) develop a CAD/CAM system

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

AD

AVRADCOM Report No. 78-29

Production Engineering Measures Program
Manufacturing Methods and Technology

COMPUTER-AIDED DESIGN AND MANUFACTURING FOR EXTRUSION OF ALUMINUM, TITANIUM AND STEEL STRUCTURAL PARTS

PHASE II - APPLICATION TO PRACTICAL EXTRUSIONS

VIJAY NAGPAL, CARL F. BILLHARDT AND TAYLAN ALTAN
BATTELLE, Columbus Laboratories
505 King Avenue
Columbus, Ohio 43201

January 1978

AMMRC TR 78-26

Final Report Contract Number DAAG46-75-C-0054

VOLUME II

Approved for public release; distribution unlimited.

Prepared for
U.S. ARMY AVIATION RESEARCH AND DEVELOPMENT COMMAND
St. Louis, Missouri 63166

ARMY MATERIALS AND MECHANICS RESEARCH CENTER
Watertown, Massachusetts 02172

FOREWORD

This final report on "Computer-Aided Design and Manufacturing for Extrusion of Aluminum, Titanium, and Steel Structural Parts - Phase II - Application of CAD/CAM to Practical Extrusions" covers the work performed under Contract DAAG46-75-C-0054, with Battelle's Columbus Laboratories, from July 19, 1976 to October 18, 1977.

The project was sponsored by the U.S. Army Aviation R & D Command, St. Louis, Mo. and contracted by the Army Materials and Mechanics Research Center, Watertown, Massachusetts. The U. S. AAVRADCOM project engineer was Mr. G. Gorline. The technical supervision of this work was under Mr. Roger Gagne of AMMRC.

This project has been conducted as part of the U.S. Army Aviation Manufacturing Methods and Technology Program, which has as its objective the timely establishment of manufacturing processes, techniques, or equipment to ensure the efficient production of current and future defense programs.

This program has been conducted in the Metalworking Section of Battelle's Columbus Laboratories, with Mr. T. G. Byrer, Section Manager. The principal investigators of the program are Dr. Vijay Nagpal, Research Scientist, Mr. Carl F. Billhardt, Staff Scientist, and Dr. Taylan Altan, Research Leader. Dr. Nuri Akgerman has been consulted throughout the program and contributed significantly to the quality of CAD/CAM programming effort.

In this phase of the program, a number of companies assisted Battelle in the development of CAD/CAM techniques through useful critique and discussions. The authors would like to acknowledge the assistance they received, especially from Mr. Mel F. Henley of Martin Marietta, Mr. C. O. Stockdale of ALCOA, Dr. R. J. Livak and Mr. F. S. McKeown of Consolidated Aluminum, and Mr. Jack Hockema of Kaiser.

the intuitive and empirical methods. Therefore, extrusion die design and manufacturing is still considered an art rather than a science. In this respect, the state of the art in the extrusion technology is very similar to that of other metal-forming processes. The scientific and engineering methods, successfully used in other engineering disciplines, have not been utilized in extrusion. This situation can be explained by the inherent complexity of the extrusion process. The difficult-to-predict metal flow, the simultaneous heat generation and transfer which takes place during the process, the friction at the material-tool interfaces, and the metallurgical variations, make the extrusion process difficult to analyze from an engineering point of view. However, recently, computer-aided techniques for analyzing and simulating metal flow and deformation mechanics have been developed and proven. The application of these techniques along with advanced numerical machining (NC) allows the practical use of CAD/CAM in extrusion technology.

Program Approach

The Phase-II work included the following major tasks:

(1) Assemble geometric modules to represent practical extrusion shapes.
(2) Apply the CAD/CAM method to a streamlined die.
(3) Develop a CAD/CAM system for nonlubricated extrusion through flat-face dies.
(4) Conduct extrusion trials and evaluate CAD/CAM extrusion results.
(5) Evaluate the economics of CAD/CAM in extrusion.

Outline of the Final Report (Phase II)

Following the major steps identified in program approach, this final report is presented in two volumes. The first volume describes the technical work and is organized in three chapters:

- Chapter 1: CAD/CAM of Streamlined Dies for Lubricated Extrusion of "T" Sections
- Chapter 2: CAD/CAM of Flat-Face Dies for Nonlubricated Extrusion of Aluminum Structural Shapes
- Chapter 3: Extrusion of "T" Sections of Aluminum, Titanium and Steel using Computer-Aided Techniques

Volume 2 is the Instruction Manual and describes the content and the use of the system of computer programs ALEXTR. ALEXTR is the CAD/CAM system developed for nonlubricated extrusion of aluminum structural shapes.

In Volume 1, each chapter can be read separately, without having to go through the entire report to find information related to any of the major tasks conducted in this program. Thus, the use and the readability of this volume of the final report is enhanced.

Chapter 1 of Volume 1 of this final report describes the application of computer-aided manufacturing (CAM) and numerical control (NC) machining techniques to the manufacture of dies for extruding shapes such as L's and T's. To illustrate the application, lubricated extrusion of "T" shape was considered. Wood models of the EDM electrode for manufacturing the streamlined dies were machined by NC techniques developed in this phase of the work.

Chapter 2 of Volume 1 describes the work conducted towards applying CAD/CAM techniques to the nonlubricated extrusion of structural shapes of high-strength aluminum alloys. A system of programs called "ALEXTR" was developed for designing flat-face dies for extruding aluminum structural shapes, and for manufacturing the flat-face dies via numerical control (NC) machining techniques.

Chapter 3 of Volume 1 describes the extrusion trials conducted to evaluate the CAD/CAM techniques developed in this program. "Tee" sections of aluminum alloy Al 7075, titanium alloy Ti-6Al-4V, and AISI 4340 were extruded using flat-face and streamlined dies designed and manufactured using "ALEXTR" and "SHAPE" systems of computer programs. The results show the validity of CAD/CAM techniques developed under this program.

Volume 2 is the User's Manual which describes the use of "ALEXTR" system of computer programs for designing and manufacturing flat-face dies.

RESULTS OF PHASE-II EFFORT

The overall objective of this program was to develop practical computer-aided design and manufacturing (CAD/CAM) techniques for extrusion of aluminum alloys, steels and titanium alloys. The objective was completed by the Phase-II effort which utilized, as foundation, the work done in Phase I. Two interactive CAD/CAM systems, namely, "SHAPE" and "ALEXTR", have been developed. "SHAPE" system of computer programs allows CAD/CAM of streamlined dies for lubricated extrusion process which includes extrusion of steel and titanium alloys. "ALEXTR" computer system allows CAD/CAM of flat-face dies for the nonlubricated extrusion of aluminum alloys.

Phase II results can be summarized as follows:

- The computer-aided manufacturing and NC techniques developed in Phase I were extended to make "SHAPE" computer system applicable to extrusion of structural shapes, such as L's and T's.
- "ALEXTR" computer system for nonlubricated extrusion of aluminum alloys was developed.
- An economical-technical evaluation of the usefulness of CAD/CAM system in extrusion was performed. This study showed that utilization of CAD/CAM systems developed in this program can reduce manufacturing costs, improve delivery schedules, and increase the productivity of extrusion operations.

VOLUME 2

COMPUTER-AIDED DESIGN AND MANUFACTURING FOR EXTRUSION OF ALUMINUM TITANIUM AND STEEL STRUCTURAL PARTS PHASE II - APPLICATION TO PRACTICAL EXTRUSIONS

USER'S MANUAL

for

"ALEXTR" SYSTEM OF COMPUTER PROGRAMS FOR DESIGN AND MANUFACTURE OF FLAT-FACE DIES FOR ALUMINUM EXTRUSION

Carl F. Billhardt
Vijay Nagpal
Taylan Altan

BATTELLE
Columbus Laboratories
505 King Avenue
Columbus, Ohio 43201

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS
(Continued)

USER'S MANUAL

for

"ALEXTR" SYSTEM OF COMPUTER PROGRAMS FOR DESIGN AND MANUFACTURE OF FLAT-FACE DIES FOR ALUMINUM EXTRUSION

INTRODUCTION

Extrusion is a process in which material is forced to flow through a shaped opening, thereby producing a length of product of constant cross section. A billet, usually cylindrical in form, is placed in a container. A die is positioned at one end of the container. This die has one or more openings in it, each opening being similar to, but not exactly the same as the shape to be extruded. By pressing on the billet usually on the end opposite the die, the billet is forced to flow through the die openings.

Aluminum extrusions are used in a wide variety of military hardware. Wing spars, fuselage stringers, helicopter blades, and airfield landing mats are typical extruded products. Extrusions offer significant advantages over other metalworking operations, especially where considerable length of the same cross-section design is desired. Extrusions can be produced to very close tolerances, thereby requiring little or no subsequent machining. They can be produced to virtually any length desired. Typically, extrusions are produced in long lengths (i.e., 100 ft) and then saw cut to the exact length desired by the customer. Extrusions can also be designed to mechanically interlock with other pieces, thereby reducing or eliminating the need for separate fasteners and additional labor.

As noted earlier, the dies used to produce aluminum extrusions are very similar to, but not exactly the same as the extrusion to be produced. This is because dies deflect under the extrusion load, thereby changing the dimensions of the opening. The die designer must anticipate these changes and design the die so that the finished product produced is what is desired, regardless of how different the unloaded die opening is from the finished product.

To assist a designer in making a die design which will produce a given finished product, a set of computer programs, ALEXTR and EXTCAM, have been written. ALEXTR assists the designer in determining:

- The optimum number of openings to have in a die
- The location of the openings of the die
- The stresses in the die and support tools
- The compensation of the die openings for deflection under load
- The bearing dimensions for balanced metal flow
- Thermal shrinkage of the extrusion and thinning during stretching.

EXTCAM uses the data developed in ALEXTR to produce other sets of data for numerically-controlled (NC) machining of the die. EXTCAM produces NC tapes for the following operations:

- Machining the extrusion template
- Machining the die template or front EDM electrode
- Machining the die opening from front or back directly into the die
- Machining the die bearings into the back of the die

The purpose of this User's Manual is twofold. The first purpose is to familiarize a die designer with the features and operation of the ALEXTR and EXTCAM programs. Both programs are written so as to operate interactively with the user. ALEXTR can operate only in an interactive mode. However, EXTCAM could be easily modified to operate in a batch mode, with all input data punched on cards. ALEXTR often requires the user to input information by "pointing" with a light pen to features of an extrusion image on a cathode ray tube (CRT) display device. This requires that the designer be knowledgeable in the trade and that ALEXTR be used as a high-speed clerical assistant. As such, ALEXTR will carry out detail calculations where specified, and will present the effects of various options. It remains with the user, however, to make decisions where more than one possible option exists.

The second function of this manual is to provide guidance to a systems analyst as to the structure, program details, and implementation of ALEXTR and EXTCAM. The information is provided to allow an analyst to learn the internal workings of the programs. This could be used to allow the programs to be modified to meet conditions unique to a particular facility, or to allow new features to be added.

HARDWARE OVERVIEW

The ALEXTR die design system is meant to be operated by an extrusion die designer in a way which complements his own skill and experience. The use of the system requires no prior experience or familiarity with computers. The following paragraphs provide an overview of the hardware on which ALEXTR runs. A detailed description of the hardware is given in Appendix III.

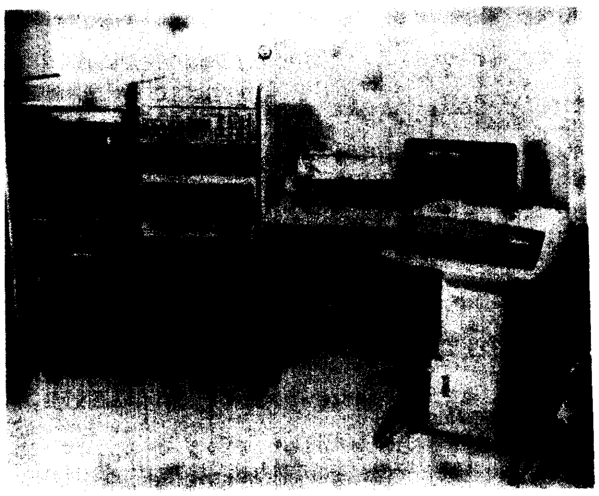


FIGURE 1. PDP-11/40 MINICOMPUTER SYSTEM WITH REFRESH GRAPHICS DISPLAY TERMINAL USED IN DEVELOPING THE "ALEXTR" SYSTEM OF COMPUTER PROGRAMS

The hardware is shown in Figure 1. It consists of the following major components:

- Two side by side cabinets containing the central processing unit (the actual "computer"), the disk drives for storage of programs and data, and the paper tape punch.
- A keyboard terminal. The user enters data values or his choices to yes/no type questions through this terminal. ALEXTR always supplies a prompting message before waiting

for the user to respond. By selecting the proper combination of keys, output from ALEXTR, such as prompting messages and results, may or may not be typed on the terminal. These messages are always printed on the CRT.

- Cathode Ray Tube (CRT). The CRT is a television-like device on which text messages may be printed, and graphic figures drawn. At various points in ALEXTR, the user interacts with the CRT display via the light pen. The light pen is a pen-like device connected to the CRT by means of a small cable. An example of the interaction of the light pen and the CRT is when an alloy is to be selected. The alloys are listed on the CRT using their four-digit designators. To choose a particular alloy, the user touches the light pen to the proper designator on the CRT. When the CRT senses the light pen, the designator will start to blink. The program will proceed after the light pen is next touched to the word "ACCEPT" which is also displayed on the CRT.
- An X-Y plotter. At various points in ALEXTR, the user may elect to make a hard-copy plot of a display on the CRT. The hard copy plot is produced on the X-Y recorder.
- A line printer (not shown in Figure 1). This may be used to get output listings of various results generated by ALEXTR. These results are saved on a disk file as part of the normal operation of ALEXTR. If the user chooses, this file may be listed on the printer using a utility program supplied by the computer manufacturer.

OPERATION AND USE OF ALEXTR EXTRUSION DIE DESIGN SYSTEM

ALEXTR is a system of computer programs which operates interactively with a user to produce designs of flat-face dies for the extrusion of aluminum shapes. The system uses a graphics display unit to show the extrusion section and die layout to the user. This gives the user immediate, visual feedback as to what the programs are doing. The system will arrange the openings of a multi-hole die based on simple design rules. The user has complete freedom, however, to modify the arrangement. This is done by identifying the particular hole to be changed with the light pen, and then entering the parameters to be changed through the keyboard.

The following conventions have been used to structure the ALEXTR system:

(1) ANSI FORTRAN IV standards are followed as much as is practical.

(2) The extrusion shapes are described in a cartesian coordinate system. A shape is defined as a polygon of X, Y points, each point having a radius of some value associated with it. Any point may be used as the starting point; the points, however, must be in clockwise order. A typical extrusion input polygon is shown in Figure 2.

(3) The program is intended to operate interactively with a designer using a Cathode Ray Tube (CRT) display terminal. The program is not intended to be used in a batch mode, non-interactive manner.

An overview of the program operation is given in Figure 3. The interaction between the user and the computer is shown in Figure 4. Figure 5 shows some of the specifics of the user/program dialog. Figures 6 through 8 are copies of the images displayed on the CRT at the corresponding points noted in Figure 5.

When started, the user must enter two file names. The first is the print file, to which is sent a copy of all output sent to the CRT. This file may be listed on the line printer when the user is through with ALEXTR. The second file is for the input data describing the extrusion section.

The user is next asked to enter the section identification number. This query is typical of most in that a default value is allowed. The default value is used when a carriage return (CR) is typed on the keyboard without entering any value. The default value for the section number will be the next section in the data file. If a positive number is entered, the file will be searched from the current section forward until a match is found between the value entered and a section ID number in the file. If a negative value is entered, the data file is rewound to the beginning before the search is started.

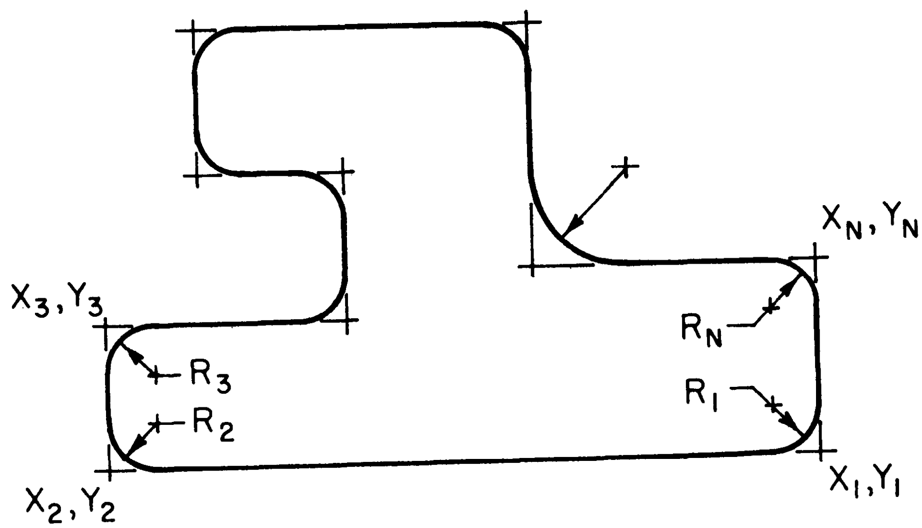


FIGURE 2. POLYGONAL DEFINITION OF EXTRUSION SHAPE

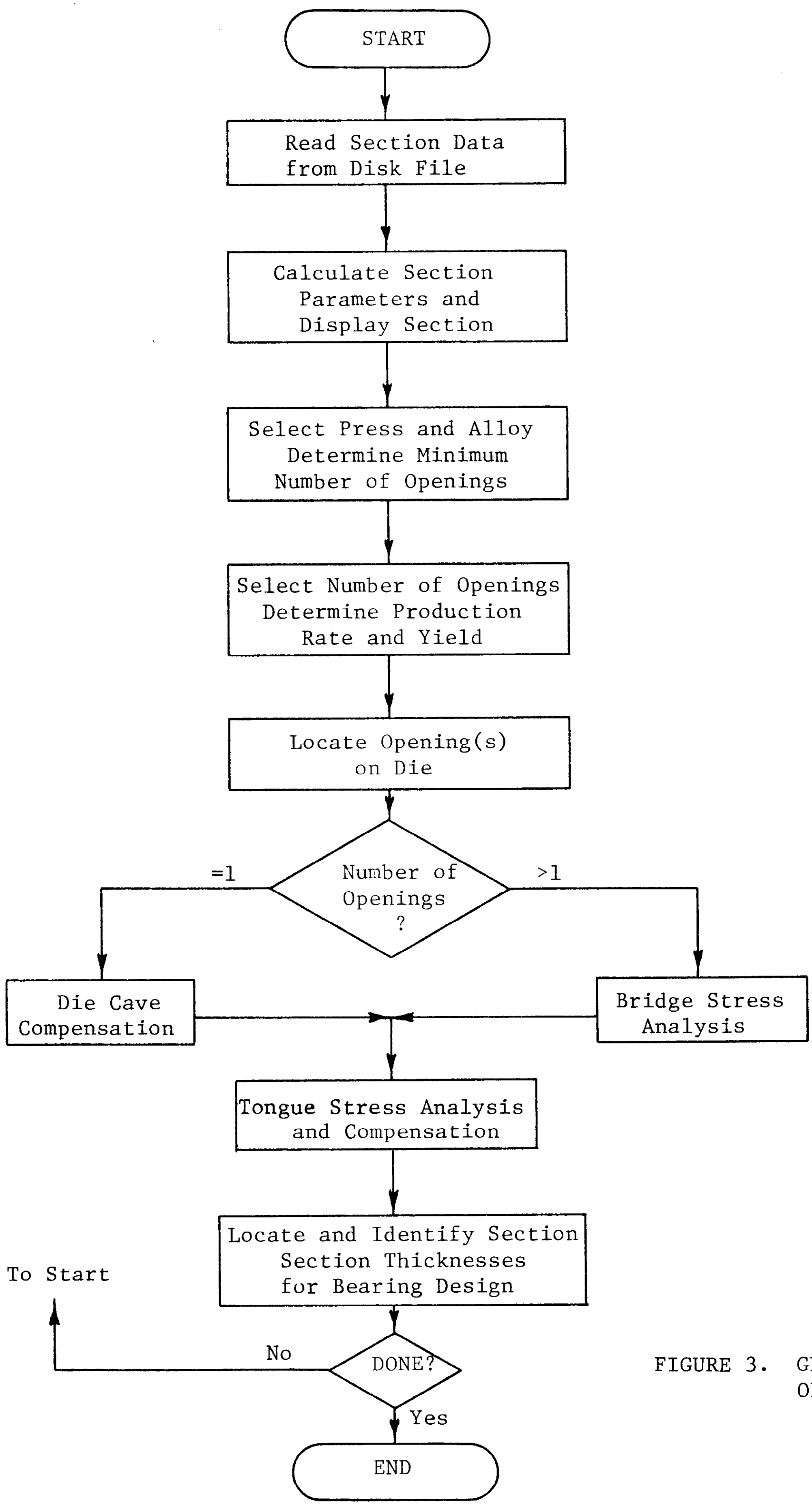


FIGURE 3. GENERAL OPERATION OF ALEXTR

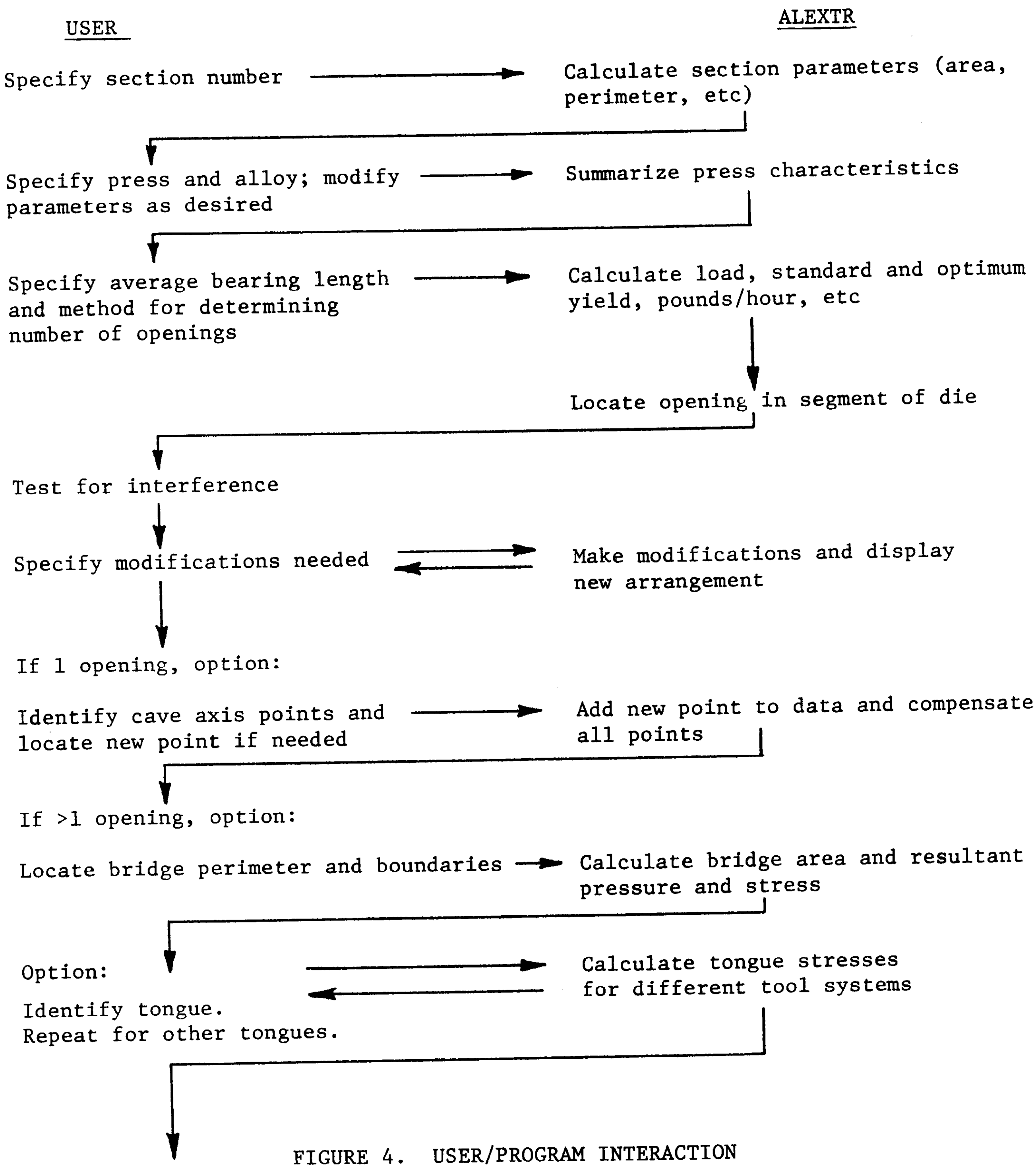


FIGURE 4. USER/PROGRAM INTERACTION

USER		ALEXTR
Identify 3 points for each section thicknesses	→	Determine section thickness
Locate bearing transition points	→	Format bearing transition and size data
Specify disk file for die design data; Specify next section number		

FIGURE 4. (Continued)

```
. R ALEXTR

PRINT FILE NAME? *TEST.PRT

DATA FILE NAME? *EXTRUS.DAT

SECTION NUMBER?

SEND DETAILS TO PRINTER (Y/N)?


SECTION NO.    1
CROSS-SECTION AREA =   0.441  PERIMETER =    3.79  SHAPE FACTOR    7.16
COORDINATES OF CENTROID ARE X =   0.337    Y =   0.337
CIRCUMSCRIBING CIRCLE DIAMETER   1.36  CENTER AT X =  0.497,   Y =  0.494
HARD COPY PLOT (Y/N)? Y
 ANY ADJUSTMENTS?
```

(See Figure 6)

```
PRESS NUMER? 2

INDICATE & ACCEPT ALLOY


MATERIAL FLOW STRESS ( 3250.)?

EXTRUSION SPEED (135.)?

RUNOUT LENGTH (120)?

LENGTH MULTIPLE (16)?

PRESS CYCLE TIME - SECONDS ( 45.)?

CONTAINER DIAMETER (10.000)?

BILLET DIAMETER ( 9.750)?


PRESS SYSTEM NO.  2, CAPACITY (TONS): 3000., CONTAINER DIAMETER: 10.000
BILLET -- DIAM:  9.750,  MAX LENGTH: 36.0,  AREA:  74.7,  WEIGHT: 268.8
BUTT LENGTH:  2.0,  RUN-OUT LENGTH (FT): 120,  MAX. NO. OF OPENINGS:  8
SPEED (FPM): 135., CYCLE TIME (SECS): 45.,  EXTRUSION RATIO: 178.1
PAUSE

AVERAGE BEARING LENGTH (0.125)?


MINIMUM NUMBER OF OPENINGS:  1, MAXIMUM LOAD (TONS):  2083


MINIMUM EXTRUSION LENGTH (24)?

METHOD FOR NUMBER OF HOLES:
OPERATOR(1), MAX LENGTH(2), MAX YIELD(3)? 1

HOW MANY HOLES (MIN =   1, MAX =   8)? 4
```

FIGURE 5. TYPICAL USER/SYSTEM DIALOG. USER'S RESPONSES ARE UNDERLINED. WHERE NO RESPONSE IS SHOWN, DEFAULT VALUE IS USED.

```
WITH STANDARD BILLET, 4 HOLES, AND 104 FOOT EXTRUDED LENGTH,
RECOVER RATIO IS 75.6%, EXTRUSION RATIO:  44.5
THIS GIVES  6 - 16 FOOT MULTIPLES, AND  8 FOOT LOSS PER HOLES.
PAUSE --


WITH 135. FPM EXTRUSION SPEED, AND  45 SEC. CYCLE TIME,
 39.5 PRESS CYCLES PER HOUR CAN BE RUN.
THIS GIVES 10607. BILLET POUNDS AND   8020. EXTRUDED POUNDS PER HOUR.
PAUSE--


BEST YIELD WITH SPECIFIED OPENINGS
NUMBER OF HOLES:  4, EXTRUDED LENGTH: 104
BILLET LENGTH: 31.6, RECOVERY RATIO: 86.2%, EXTRUSION RATIO:  44.5
 6 - 16 FOOT PIECES PER HOLE
PAUSE --


WITH   4 OPENINGS, THE BREAKTHROUGH LOAD IS   1870. TONS.


NEW SECTION(1), NEW PRESS(2), LOAD(3), LOCATE(4), DONE (5)? 4


TOLERANCE (0.750)?

HARD COPY PLOT (Y/N)? Y    (See Figure 6)
 ANY ADJUSTMENTS?

POSITION OK (Y/N)? Y

HARD COPY PLOT (Y/N)? Y    (See Figure 7)
 ANY ADJUSTMENTS

POSITION OK (Y/N)
```

*

FIGURE 5. (Continued)

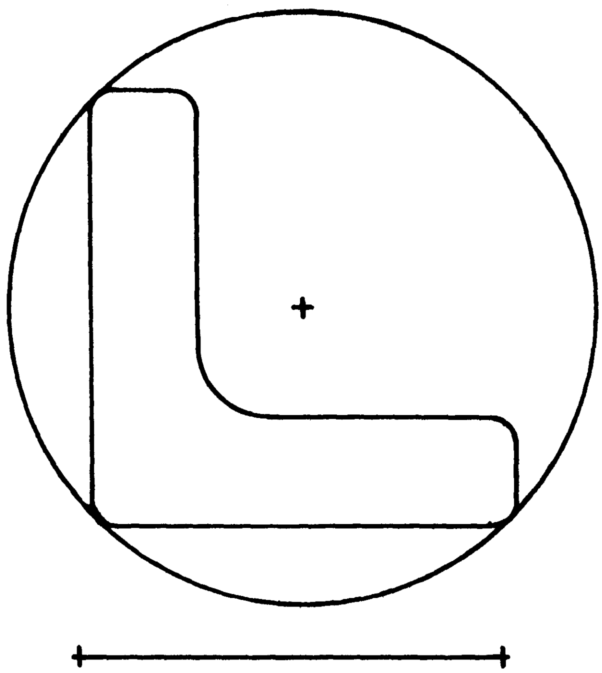


FIGURE 6. EXTRUSION SECTION AND CIRCUMSCRIBING CIRCLE
(line at bottom is 1-inch unit vector)

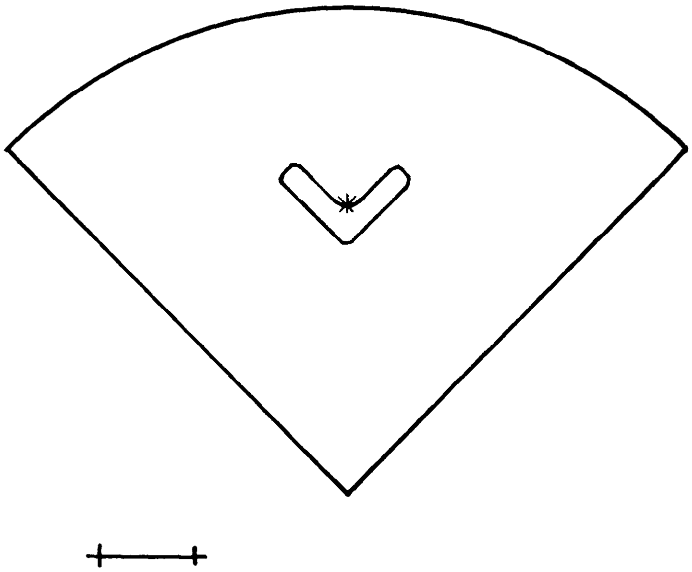


FIGURE 7. EXTRUSION SECTION PLACED IN ONE-FOURTH SEGMENT OF DIE

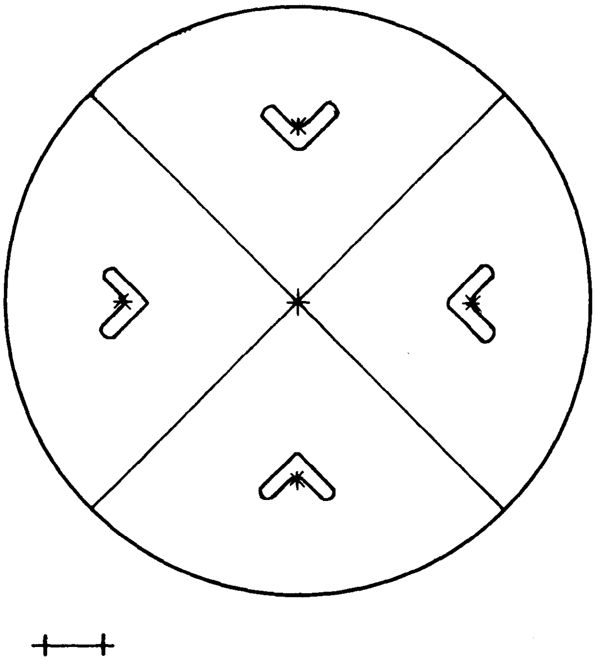


FIGURE 8. ARRANGEMENT OF ENTIRE DIE BY REPEATING LAYOUT OF ONE SEGMENT FROM FIGURE 5

As shown in Figure 5, some queries indicate the numeric value of the default response. If no new value is entered, the value shown is used. In the case of Yes/No queries, N (no) is always the default value.

After the section number is specified, the user is asked if details are to be sent to the printer. If this is requested, details of various intermediate results are sent directly to the printer. These details are usually used only for debugging purposes and would not normally be of interest to a die designer.

ALEXTR then proceeds to calculate and report the section parameters. The values printed include area, perimeter, shape factor (perimeter/weight/foot), location of centroid, and the size and location of the circumscribing circle. The extrusion shape and the circumscribing circle and its center are displayed on the CRT at the time the geometric parameter values are printed. Figure 6 was made by requesting a hard copy of this display.

The user is then asked to enter the press system number to be used. There is no default value for this query. At present, characteristics for three presses have been incorporated into ALEXTR. Therefore, the value 1, 2 or 3 must be entered. Any other response will cause the question to be repeated. The data for these presses was made up by Battelle and is meant to represent typical press configurations. An actual user would have to reconfigure the data in the tables to describe his own presses.

When a valid press number is entered, the user is asked to identify the alloy to be used. The alloys are displayed as four-digit light buttons (i.e., 1100, 2024, etc.) As a light button alloy code is touched with the light pen, the light button will start to blink to verify the light pen hit. After selecting the proper alloy, the user must touch the light button "ACCEPT" to have the alloy choice registered. Four properties are associated with each alloy: the flow stress, nominal extrusion speed, the coefficient of thermal expansion, and the nominal extrusion temperature. The first two are used in subsequent calculations to determine the expected load and production rate. The second two are used in EXTCAM to determine the expansion factor for the die. The user must then accept the default values, or enter new values for the material flow stress and extrusion speed, and the press runout length, cycle time, container size, and the finished length multiple. When these questions are answered, the characteristics of this press are summarized.

Load and yield calculations are next made with the first step being a query to the user as to the average bearing length of the die. This value, plus the section and container geometry are used to predict the maximum load. Since the load decreases as the number of openings increases, the program makes the first load calculation based on a single opening. If the result exceeds the press capacity, the calculation is repeated until the expected load is less than the press capacity or the number of openings required is greater than the number allowed by the defined press characteristics.

The minimum extrusion length is then requested. The default value shown is the length multiple entered previously plus 8 feet. The 8 foot added is a defined amount of loss for die breakthrough and stretching grips. The minimum extrusion length entered may not be less than the default value nor greater than the runout length.

The user must then choose the method for selecting the number of openings in the die. This is done by typing the appropriate number if response to the question

"OPERATOR(1), MAX LENGTH(2), MAX YIELD(3)".

If 1 is entered, the user specifies the number of openings. ALEXTR then finds the best material yield based on this number of openings. If 2 is entered, the best material yield is found based on the criteria that the extruded length is to be as long as allowed by the runout table. If 3 is entered, the number of openings needed for maximum yield from a standard billet is calculated. In addition the number of openings required to produce the greatest production rate (lbs/hr) is found.

With the load and number of openings determined, the next step is to position the openings on the die. As the first step in the process, ALEXTR draws the segment appropriate for the number of openings (a circle for a 1 hole die) and places the opening on the segment such that the center of gravity (CG) of the opening and the CG of the segment coincide. The opening is also rotated such that its greatest distance is parallel to and as close as possible to the chord of the die circle. This is shown in Figure 7 for a die with four openings. Had this been a 2-opening die, the "L" shaped opening would have been rotated an additional

180 degrees, thus facing downward toward the die diameter (the chord of a 180 degree segment). After the segment and its opening are displayed, a tolerance value is requested. This value is the clearance to be tested for between the opening and the container diameter, or between adjacent openings. When the tolerance is entered, or the default value accepted, two small concentric circles are drawn in the lower right corner of the CRT. The outer circle diameter is scaled to the tolerance specified and the inner circle is half this amount. This tolerance circle is light-pen sensitive. It may be positioned anywhere on the screen by the user touching the pen to the circle and then moving the pen as desired. When the pen is removed from the screen, the circle will remain at its last position. The tolerance check is ended by touching the pen to the light button text "END".

If the position of the opening is acceptable to the user, the single segment display is replaced with a display of the full container circle and all openings. Each opening is located on its segment at the same relative position as the single opening was. This is shown in Figure 8. If the positions of the several openings are acceptable, ALEXTR proceeds to cave (dish) compensation for a single-hole die, or bridge stress analysis for multi-hole dies.

The user has two opportunities to modify the position of the opening on the die. The first is while the single segment and opening are displayed. If the user uses the default value (or enters N) to the query "Position OK?", the following actions are taken:

(1) The current position of the CG of the opening relative to the center of the die is typed as:

   a) X,Y position relative to center of die

   b) Rotation, in degrees, relative to the original position

   c) Mirror image marker (1 = not mirrored, -1 = mirrored)

(2) The user is requested to indicate what action is to be taken by typing the appropriate response to the query: TRANSLATE(1), ROTATE(2), MIRROR(3), TEST(4), DONE(5).

An example of the user/system dialog to change the position of the opening is given in Figure 9. The result of this dialog was to change Figure 7 into Figure 10. It should be noted that if the translate or rotate options are selected, the user must also enter the appropriate increments in x and y coordinates (dx,dy), or the increment in rotation angle ($d\theta$), respectively. The user enters the values as incremental amounts from the current position. The values displayed, however, are absolute amounts relative to the original position. If "mirror" is requested, the original opening definition is mirrored about the Y axis through the CG and then the opening is translated and rotated as last specified. If "test" is requested, the tolerance value is requested and then the L.P. (light pen) sensitive tolerance circle is generated and used as previously described. "Done" will terminate the position manipulation process.

While in the position manipulation process, there are two ways to translate or rotate an opening. The first is to specify the incremental amount in inches by which the opening is to be translated along each axis, or the incremental amount in degrees that the opening is to be rotated. The second method of translation or rotation is through the use of the light pen. This works as follows for translation. If the user requests translation, but enters Ø, Ø for the incremental move in X and Y, a tracking mark will be displayed on the CRT, located at the center of gravity of the opening. By touching the tracking mark with the light pen and then moving the light pen about the screen, the position of the opening can be moved freehand. When the light pen is removed from the CRT, the opening will remain where last positioned. This freehand mode of translation is terminated by touching the light pen to the "END" light button.

If rotation is specified and Ø is entered for the incremental angle to move, the polygon coordinates describing the opening are displayed as points. By selecting any two points with the light pen, the opening will be rotated such that the line between the two points is made horizontal.

When the single opening is positioned as desired, the multi-opening display is generated by repeating the pattern for the first opening. Thus, Figure 11 follows from 10. This may be accepted as is, or the positions of the individual openings may be modified. This would be done in a similar manner to that described above. However, the item to be modified must first be identified. This is done by touching the desired item with the light pen and then touching the "ACCEPT" light button. When an opening is touched with the L.P., the image will start to blink. If another opening is then touched, the first will stop blinking and the second will start blinking. When an item is finally "ACCEPT"ed, it will stop blinking but its CG marker will continue to blink to remind the user which item is being modified. It should be noted that the CG of each segment is marked with a "+"; the CG of each opening is marked with an "X".

```
TOLERANCE (0.750)?

HARD COPY PLOT (Y/N)?

POSITION OK (Y/N)?

 0.000   2.926      -43.           1

TRANSLATE(1), ROTATE(2), MIRROR(3), TEST(4), DONE(5)? 2

ANGLE? 178

 0.000   2.926      -45.           1

TRANSLATE(1), ROTATE(2), MIRROR(3), TEST(4), DONE(5)? 1

DX, DY? 0,1

 0.000   3.926      -45.           1

TRANSLATE(1), ROTATE(2), MIRROR(3), TEST(4), DONE(5)? 5

HARD COPY PLOT (Y/N)?
```

FIGURE 9. DIALOG USED TO CHANGE FIGURE 6 TO FIGURE 9

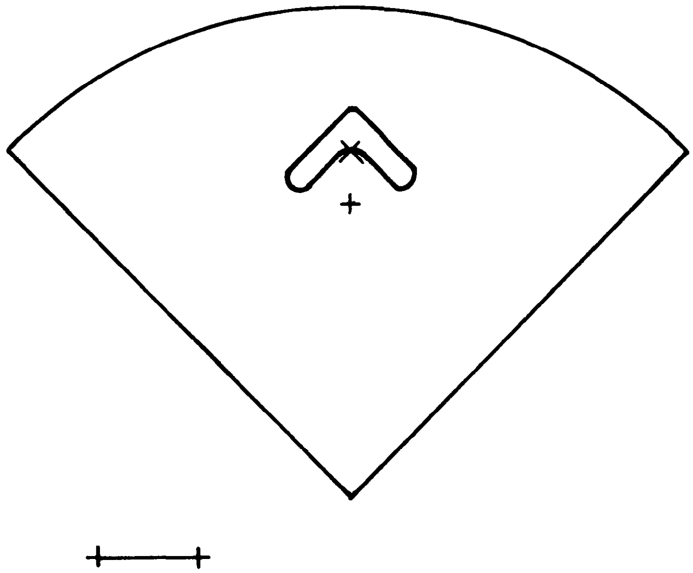


FIGURE 10. SINGLE SEGMENT LAYOUT MODIFIED BY USER'S DIRECTIVES

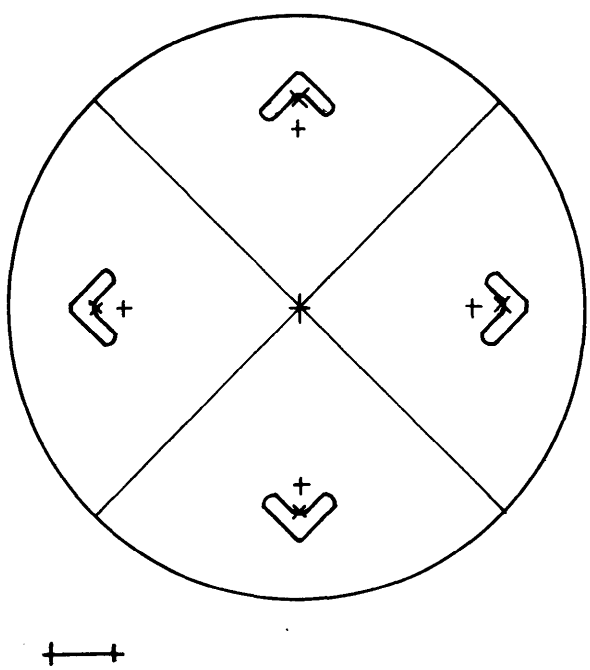


FIGURE 11. FULL DIE LAYOUT BY REPEATING SINGLE SEGMENT ARRANGEMENT FROM FIGURE 10

By repeatedly picking items and then specifying the operation, and value if appropriate, any item may be placed anywhere on the screen. Note that the system does not verify that an opening is even located on the die. This if left to the user's perception. Figure 12 is an example of modifying the location of each opening of a multi-hole die. This arrangement was derived from Figure 11 by selecting and modifying each item in various ways and amounts.

The next step in the ALEXTR die design and analysis program is dependent on the number of openings to be made in the die. If the die is to have only one opening, the user is asked if cave compensation is to be made. A die having a relatively thin opening will tend to cave, or dish, near the center of the die. To compensate a die for cave, the user first indicates two points with the L.P. The perimeter of the opening between the two points is considered to be subject to cave. That is, under load, the perimeter between the two indicated points would tend to deflect, thereby tending to close the opening.

All points between the two indicated points are shifted away from the center to compensate for the tendency of this area to close under load. However, if the two indicated points are adjacent to one another, there are no intervening points to shift. Therefore, before the points are shifted, the user is given the opportunity to add a new point. This new point is added by locating it on the opening perimeter with the L.P. In order to be valid, the new point must lie between the two previously indicated points. When the points are shifted for the cave compensation, the new point will be moved along with the original intervening points. It should be noted that if the two indicated points are immediately adjacent to one another, a new point must be added in order to make any change for cave effects.

If the die is to have more than one opening, the cave compensation process is skipped. Instead, the user may elect to make a bridge analysis. A die bridge, in the context of ALEXTR, is the area of the die which lies between two or more openings. The shaded area of Figure 13 is a typical bridge. This area is supported by the boundaries between points 1 and 20, and 10 and 11. The user locates the bridge perimeter points by using the L.P., while the display of each opening is L.P. sensitive. Once the perimeter points have been located, the points themselves are made L.P. sensitive and the openings are densitized. The user then uses the L.P. to indicate which of the bridge area points also define

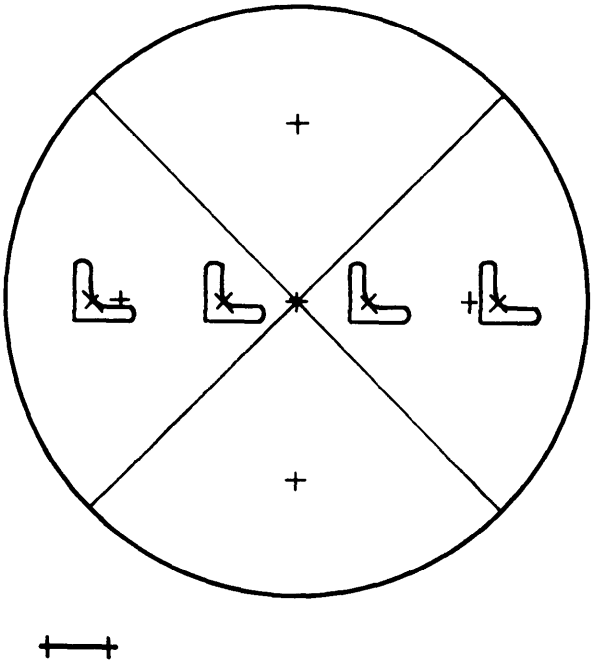


FIGURE 12. FULL DIE LAYOUT MODIFIED BY USER'S DIRECTIVES

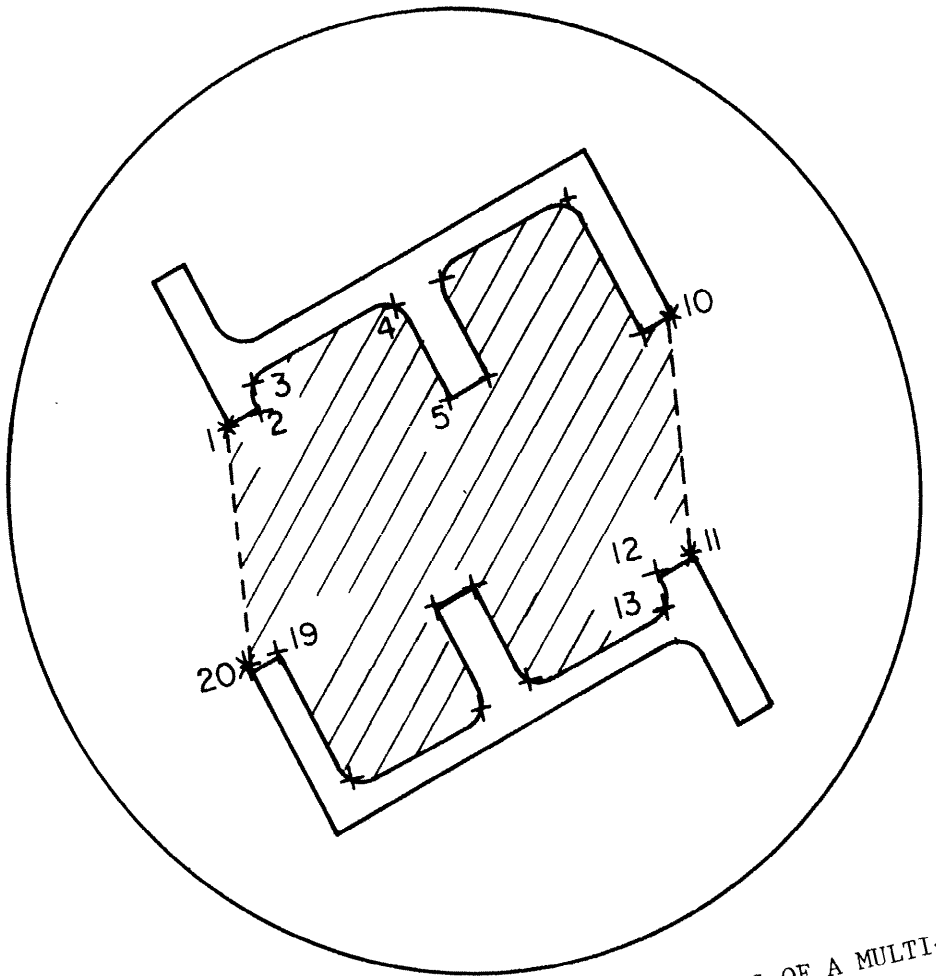


FIGURE 13. BRIDGE AREA BETWEEN OPENINGS OF A MULTI-HOLE DIE

the bridge boundary points. The surface area of the bridge is calculated along with the boundary area normal to the surface. From these areas and the expected extrusion load, the shear stresses at the boundaries can be determined. The stress calculation is made for the tool combinations of (1) die, backer and bolster, (2) die and backer, and (3) die only. The results of this calculation allows the user to evaluate the need for conforming support tools.

After the die cave compensation or bridge stress analysis is completed, depending on the number of die openings and the desire of the user, ALEXTR presents the option of compensating for tongue deflection. A tongue is an area of a die which projects into the opening such that the area is surrounded on three sides by the die opening. An example of a die tongue is that area of a die which forms the center portion of a "U". A tongue can be looked upon as a cantilever beam which will deflect under load. The deflection will cause the extrusion section at the end of tongue to be thinned. In order to make an extrusion of the proper finished dimensions, the points defining the tongue must be moved away from the tongue end.

To do this, the user indicates the first and last points of each tongue (i.e., points 2 and 5 in Figure 13). The bending and shear stresses on the area defined by all points between and including the two indicated is then calculated. This stress calculation is based on the assumption that the pressure on the die is uniform across the entire face of the die. The stresses are calculated for the same combinations of tools described above to evaluate bridge stress.

The final step in the ALEXTR extrusion die design process is for the user to indicate the thicknesses of the extrusion which are relevant to the design of the die bearings, and where on the die these thicknesses are to be applied. The die bearings, or lands, are the distances into the die across which metal flows as it is pushed through the die. The lengths of the bearings are different at different positions along the perimeter of the die opening. The longer a bearing is, the more restriction it provides to metal flow. In order to have uniform metal flow across the entire opening, the length of the bearing at any particular location is made directly proportional to the area or thickness of the opening in the vicinity of the point.

To indicate relevant section thicknesses, the user indicates a series of three points. Each set of three points is used to define a circle. The diameter of this circle is subsequently used as a section thickness. As each set of three points is indicated by the user by means of the L.P., a scaled circle is drawn through the points. When all required circles have been defined, touching the END L.B. (light button) with the pen terminates the circle definition processes.

The user then indicates the points on the perimeter between which a particular thickness is to be applied for determining the bearing length. The transition points must be selected along straight-line portions of the opening in order that the transition can be made as a ramp change rather than a step change. Starting from the first point on the perimeter (the first point is marked with an X), the user would work clockwise around the perimeter. As each transition point is located and ACCEPT'ed, the user must then indicate and ACCEPT the circle which is applicable. The diameter of the circle picked is equated to the thickness of the extrusion between the previous (or first) point and the point just located. When all bearing transition points and thicknesses have been identified, picking END with the L.P. will terminate the bearing specification process. The actual determination of the bearing lengths is a function of the distance from the center of the die to every point on the die opening and the thickness specified to be applicable at the point. This determination is made in EXTCAM when the cutter path to machine the bearings is calculated.

When the bearing specification process is completed, ALEXTR loops and asks the user:

NEW SECTION(1), NEW PRESS(2), LOAD(3), LOCATE(4), DONE(5) .

The user indicates his choice by typing the appropriate numeric response. If he indicates either a new section (1) is to be processed, or that he is finished with all ALEXTR processing (5), the user is asked if he wishes to save the die data for the section which was just processed. If he indicates Yes, he is asked to enter the name of a file in which this data will be stored. This file will be used by EXTCAM to generate the cutter paths for NC machining of the die.

Operation and Use of EXTCAM Extrusion Die Manufacturing System

EXTCAM is a series of computer programs which generate punched tapes for the NC manufacture of dies for extruding aluminum. These dies are of flat-faced design, conventional in the aluminum extrusion industry. In contrast to the ALEXTR die design system, EXTCAM makes no graphic displays on a CRT, and only interacts with the user to the extent necessary to determine what machining operation is to be calculated and what cutter size is to be used.

The operation and options of EXTCAM are shown schematically in Figure 14. When started, EXTCAM asks the user the name of the input data file to be processed. This file would have been generated as an output of ALEXTR. The die and bearing dimensions are then expanded, as required, to compensate for thermal and stretcher effects. Since aluminum extrusions are generally made from heated billets to reduce the material's flow stress and, thereby, the press load, the extrusion will shrink while cooling to ambient temperature. In order to be at size when cool, the extrusion must be made in a slightly over-size die when hot. When cool, extrusions are usually stretched slightly beyond their elastic limit in order to straighten the section. Stretching the section lengthwise will reduce the cross-sectional area. These two effects are compensated for by making the extrusion die slightly larger in the X-Y plane than the finished section dimensions.

The user is then presented the following options:

- TEMPLATE(1)
- DIE(2)
- ELECTRODE(3)
- BEARINGS(4)
- DONE(5)

By typing the numeric value associated with each of the above, calculations for the tool path which will machine the option are made. When TEMPLATE(1) is requested, the user is asked to enter the cutter size to be used. The cutter radius must be equal to, or less than the smallest fillet radius of the extrusion. The template which is cut is the template of the finished extrusion. It is not the die opening template. As part of determining the tool path, the size of the

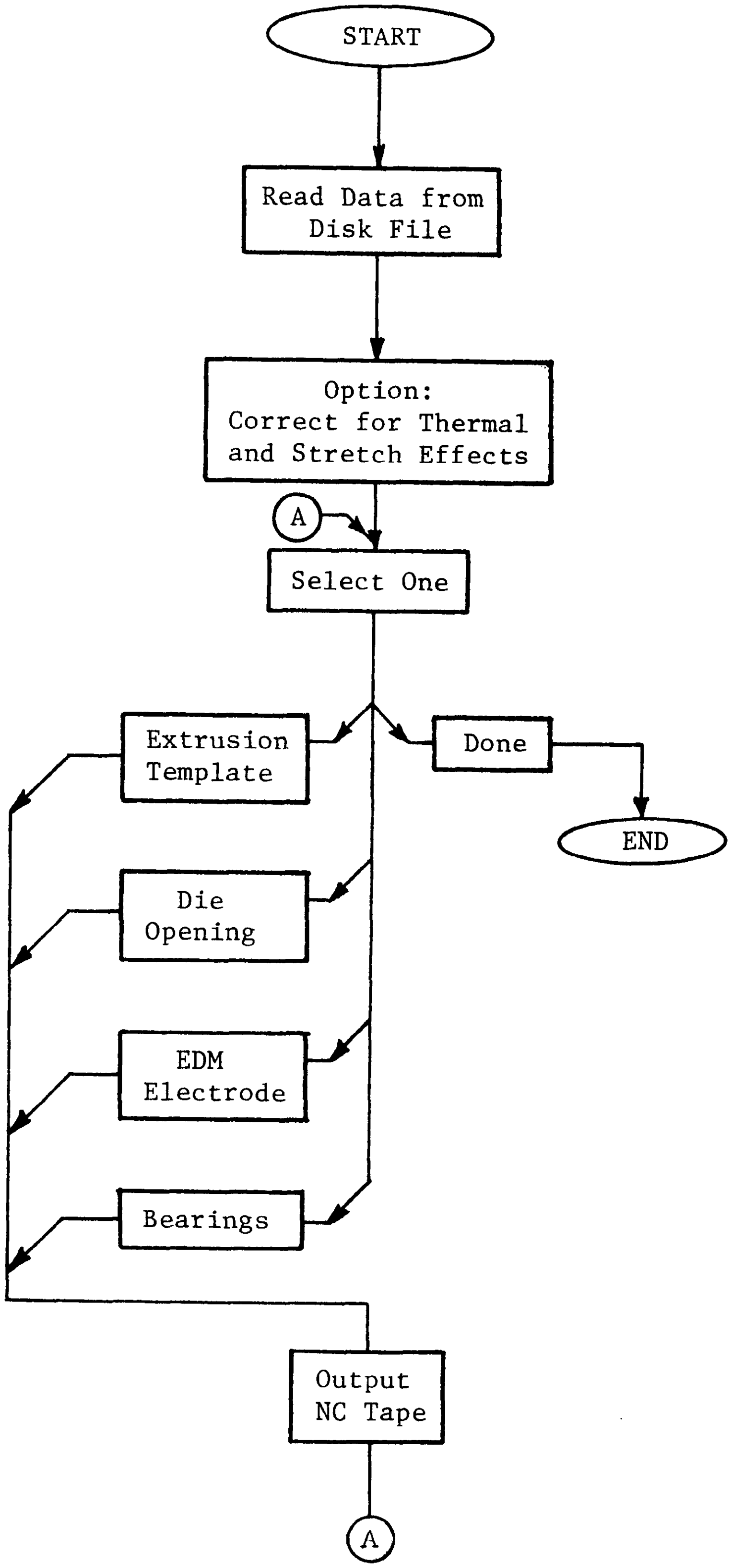


FIGURE 14. GENERAL FUNCTION OF EXTCAM

rectangular blank from which the template is to be cut is determined. The location of the initial starting point is also determined, relative to the lower left corner of the sheet blank.

The depth of cut is determined such that the cutter will be fed 0.025 inch below a 1/8-inch piece of stock when the tool origin is set 1.0 inch above the stock. The tool motion to cut the template includes a ramp down at the start and a ramp up at the end to lead the tool into and out of the stock. The tool path to cut a template is shown in Figure 15.

When the user wishes to cut a die, the tool path is calculated offset to the inside of the die polygon. This is in contrast to offsetting to the outside and using the extrusion polygon when making a template. The tool path will be calculated for the front or back of the die, and for a particular cutter size, both as specified by the user. The depth of cut is 1.0 inch plus 55 percent of the die thickness. Thus, the tool origin for the final cut would be 1.0-inch above the center of the die. The output NC tape will contain the tool positions necessary to cut, in one operation, all of the openings specified in ALEXTR. It is not necessary to reposition the origin for each opening of a multi-opening die. An example die layout is shown in Figure 16. Figure 17 shows the tool path which will cut this die from the back. When a die is to be cut, EXTCAM does not check to determine if the cutter size is valid. This can be visually verified by making a plot of the cutter path in the X-Y plane, as shown in Figure 18. If the cutter path crosses over itself, as it does in Figure 18, the cutter is too large to cut the opening.

When an electrode is specified to be cut, EXTCAM offsets the cutter to the outside of the die opening. It also creates the mirror image of each opening, since the electrode is inverted in the EDM process. The tool path to cut the electrode is given a ramp into and out of the electrode stock to prevent a dwell mark from forming while cutting to depth. All projections for a multi-hole die are cut from a single piece. The cutter motions for all projections are on the same tape, with the cutter proceeding from one projection to the next. The set point of the tool is 1.0 inch above the center of the graphite blank. At this setting, the projection will be machined 1.0-inch long. The excess material outside of the contoured projections must be removed by hand.

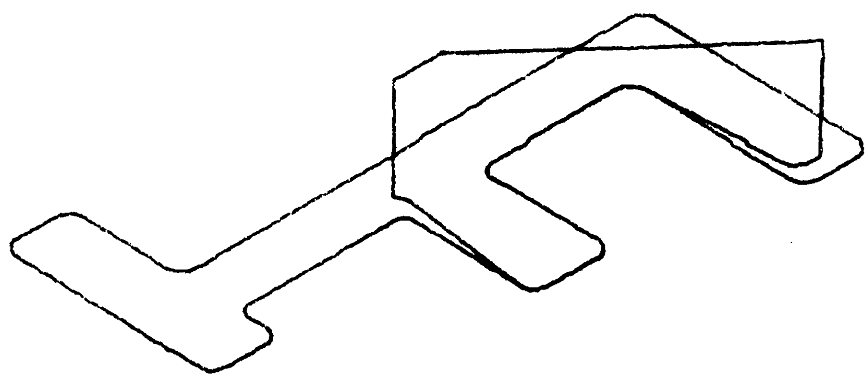


FIGURE 15. CUTTER TOOL PATH TO MACHINE A TEMPLATE

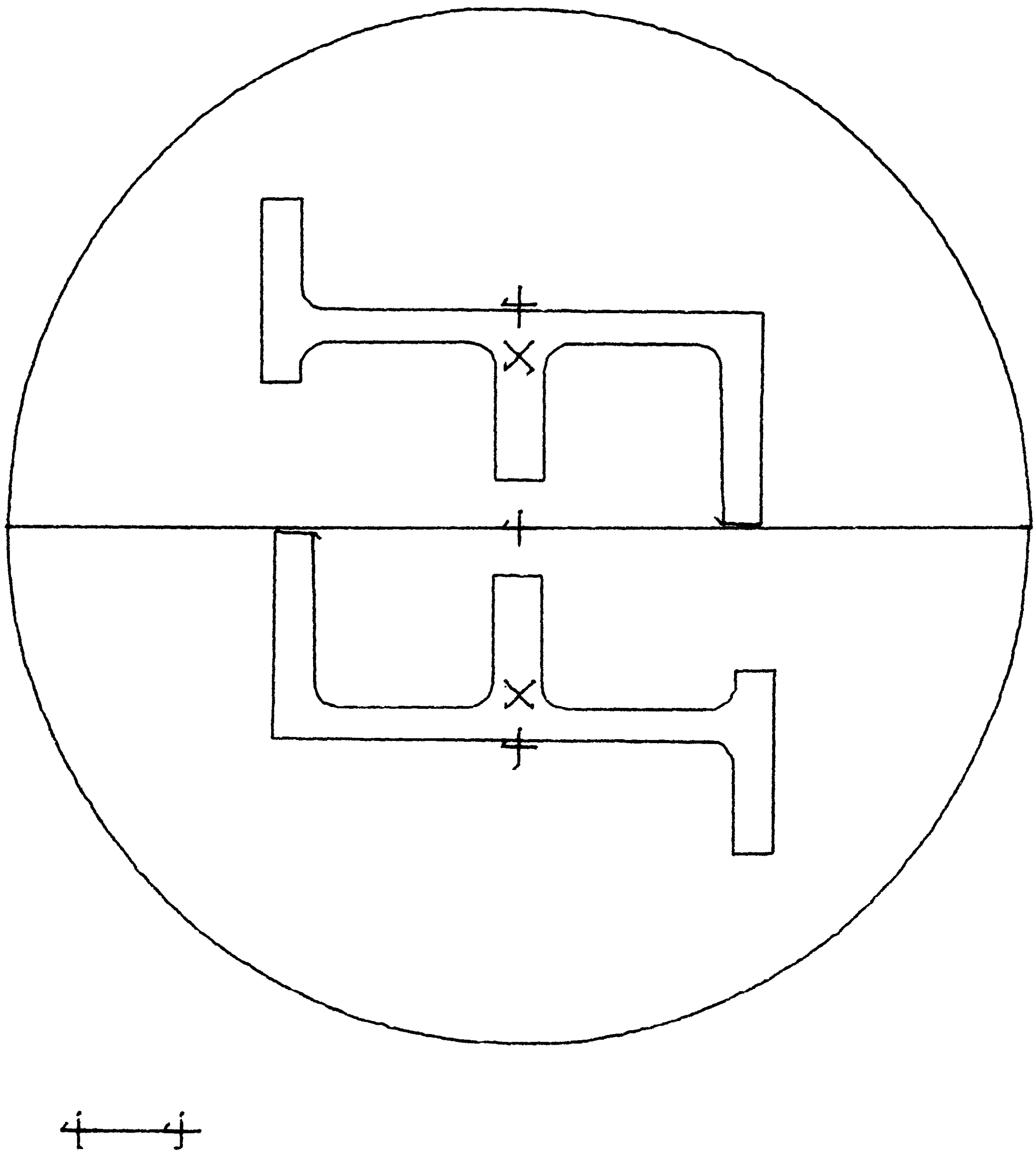


FIGURE 16. TYPICAL TWO-OPENING DIE LAYOUT

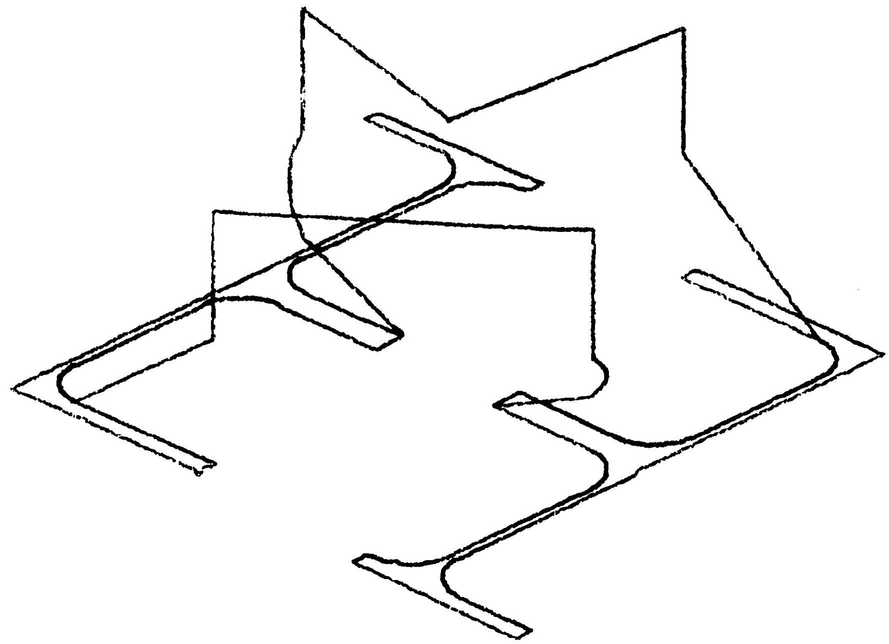


FIGURE 17. CUTTER PATH TO CUT THE DIE OPENING CORRESPONDING TO FIGURE 16 FROM THE BACK OF THE DIE

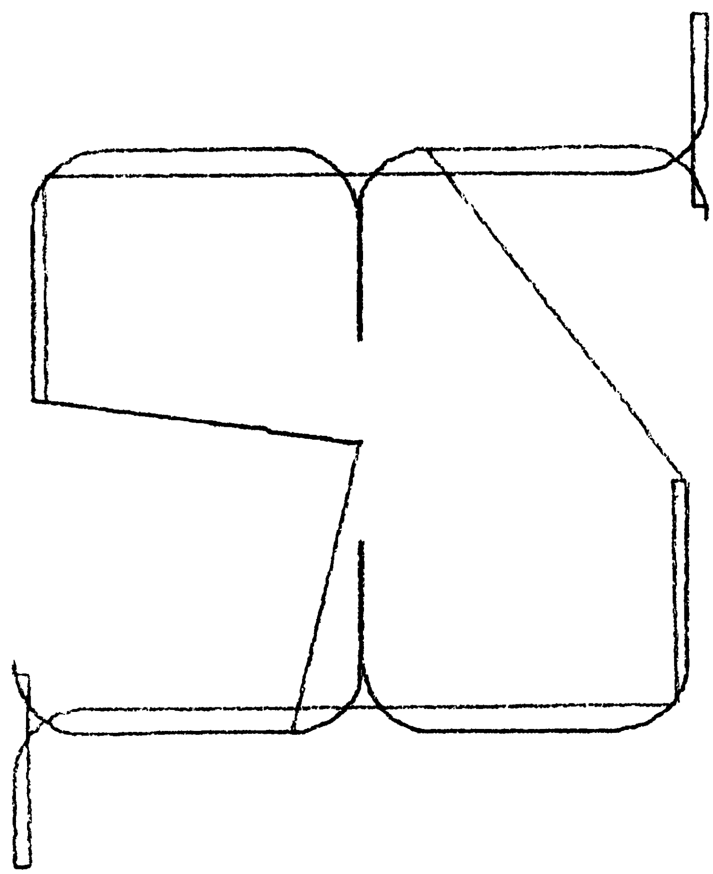


FIGURE 18. EXAMPLE OF A CUTTER PATH WHEN TOO LARGE A CUTTER SIZE WAS SPECIFIED

The final capability of EXTCAM is to generate the NC tape for machining the bearings, or die lands, into the back of the die. This uses the die polygon data, along with the bearing transition point and thickness information generated in ALEXTR. The length of the bearing at each point on the perimeter of an opening is calculated based on the thickness specified for that position, and the distance of that position from the center of the die. The closer a perimeter position is to the die center, the longer the bearing is made. This is to account for the fact that metal flow is not uniform across any cross section of the billet. Metal flow is faster at the center of the die compared to that near the container wall. This is similar to the flow of fluid in a pipe where the velocity of the fluid is a maximum at the center and decreases as one approaches the wall. Where the bearing transitions from one thickness to another, the transition is made at a 60-degree ramp. The ramp is offset, as necessary, to compensate for the cutter size.

In addition to specifying the cutter size, the user must also specify (a) the amount by which the cutter is to be offset to the inside from the true opening profile, and (b) the size of the smallest bearing. As the relative depth of all bearings is being calculated, the shortest bearing is also determined. This is then used to determine the actual depth of cut needed from the back of the die to yield the specified minimum bearing length. As with the output for cutting dies directly, or cutting electrodes, the NC tape for machining bearings will cut all openings, one after another. The cutter path for machining the bearings of a two-opening die is shown in Figure 19.

Use of the ALEXTR/EXTCAM Systems

Although a variety of sections were used to develop and test the ALEXTR and EXTCAM systems of programs, one shape was processed, manufactured and extruded. This shape is the "T" shown in Figure 20. The coordinates and the radii used to describe the polygon representing this shape are given in Table 1.

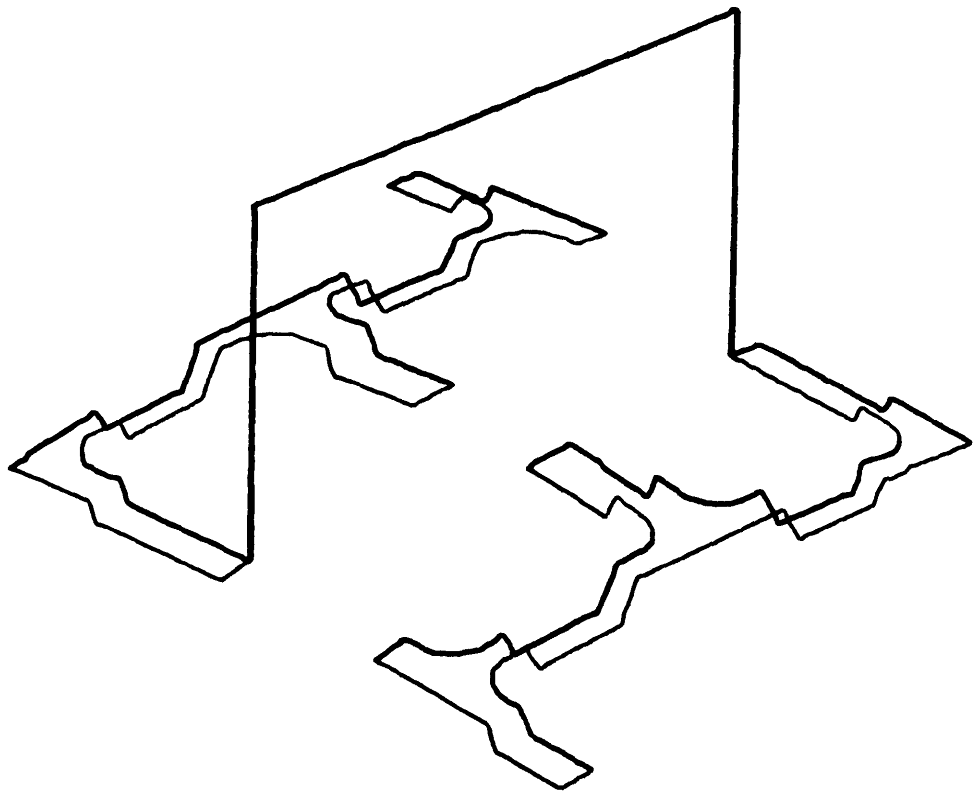


FIGURE 19. CUTTER PATH FOR BEARINGS OF A TWO-OPENING DIE

TABLE 1. "T" POLYGON COORDINATES

Point	X	Y	R
A	0.000	0.000	0.062
B	0.000	0.188	0.062
C	0.656	0.188	0.187
D	0.656	1.188	0.062
E	0.844	1.188	0.062
F	0.844	0.188	0.187
G	1.500	0.188	0.062
H	1.500	0.000	0.062

The geometric properties for this section are given in Table 2.

TABLE 2. "T" SECTION GEOMETRIC PROPERTIES

Area (in^2)	0.48
Perimeter (in)	5.06
Shape Factor	8.78
Center of Gravity (X&Y)	0.750, 0.326
Diameter-Circumscribing Circle (in)	1.63
Center-Circumscribing Circle (X&Y)	0.749, 0.373

This section was to be extruded using Battelle's 700-ton hydraulic press equipped with a 3-inch container. With an average bearing of 0.187-inch, starting billet dimensions of 2.875-inch diameter x 6.0-inch long, and flow stress of 7500 psi, a breakthrough load of 263 tons was calculated. When the average bearing was specified to be 0.250 inch, the expected load was 273 ton.

The opening was positioned in the die using the program-generated layout. That is, the center of gravity of the section was located at the center of the container. This is shown in Figure 21. The next step in the design was to provide the cave or dish compensation. This was applied to the long side of the base of the "T". Points H and A were indicated as the cave axis end points and then a new point, J, was added. The cave compensation rate was 0.004 in/in.

The stress on the tongues was then determined. The tongues were defined by points BCD and EFG (Figure 20). The tool dimensions were as follows:

- Die thickness: 1.0 inch
- Backer thickness: 1.75 inch
- Bolster thickness: 0.0 inch
- Die to backer clearance: 0.125 inch.

Although the average pressure on the die was 74,300 psi, the combined bending and shear stress was only 1573 psi on the die and 11,100 psi on the backer. If no backer was used, the tongue stress on the die would have been 40,600 psi. Thus, the decision to proceed with making the extrusion using a die and backer, but no bolster appeared reasonable.

The final step in the die design process was to indicate the section thicknesses and bearing transition points. Two thicknesses were identified. These thicknesses and the bearing transition points are shown in Figure 22. The die design data was then saved on a disk file for subsequent access by EXTCAM. The results of the ALEXTR analysis which were sent to the print file on the disk are given in Figure 23.

EXTCAM was used to make the tapes to NC machine the EDM electrodes and the die bearings. To machine the graphite electrodes, two tapes were made. The first, used for roughing, had an 0.375-inch diameter cutter specified; for the finish pass, an 0.250 cutter was specified. For both passes, an EDM burn and polish allowance of 0.002 inch was used. Also, before the electrode or bearing tapes were generated, a thermal and stretcher compensation of 1.016 in/in was applied to the die and bearing data.

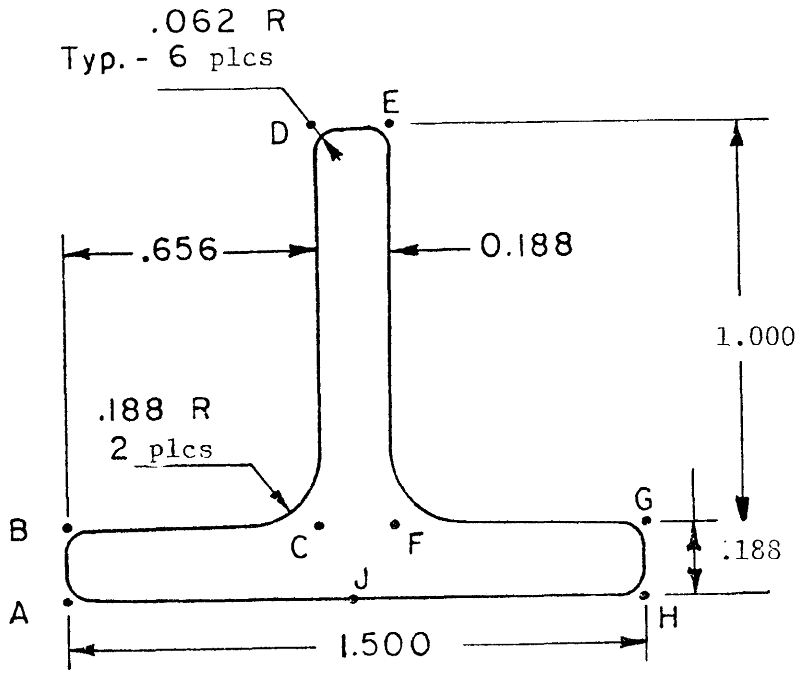


FIGURE 20. "T" SHAPE USED TO TEST ALEXTR AND EXTCAM

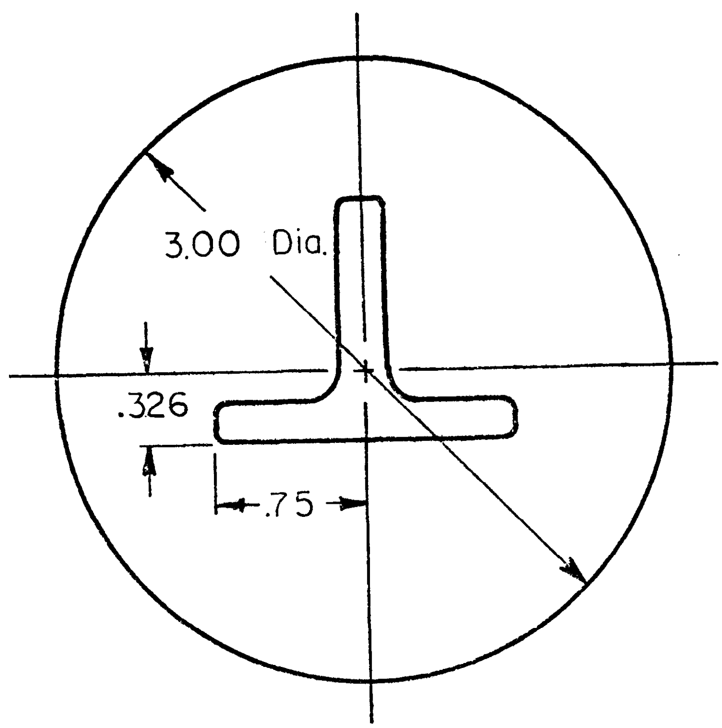


FIGURE 21. POSITION OF "T" SHAPE IN DIE USED IN BATTELLE HYDRAULIC PRESS

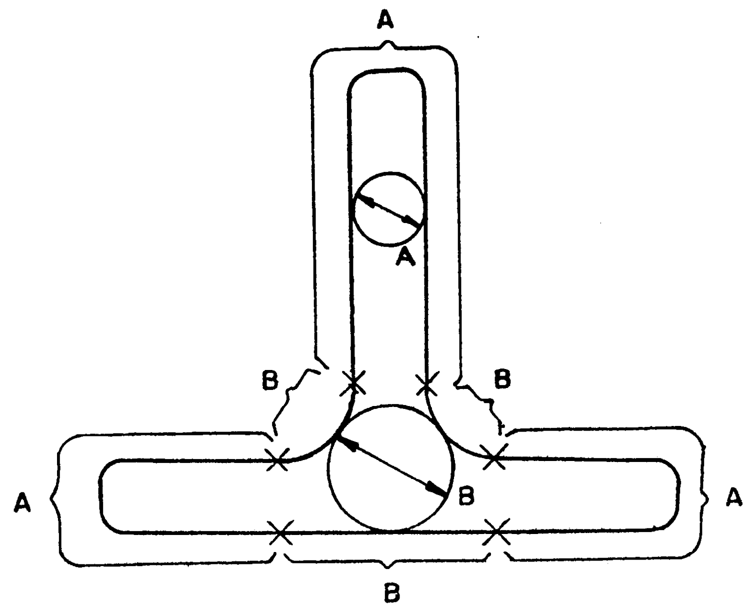


FIGURE 22. SECTION THICKNESSES AND TRANSITION POINTS USED IN THE DESIGN OF THE BEARINGS FOR "T" DIE

```
THIS IS DATA FOR BATTELLE LABS EXTRUSION TESTS


     1.5 X 1.Ø X 3/16 TEE SECTION


 SECTION NO.       2
 CROSS-SECTION AREA =    Ø.48Ø  PERIMETER =     5.Ø6  SHAPE FACTOR    8.78
 COORDINATES OF CENTROID ARE X =     Ø.75Ø      Y =    Ø.327
 CIRCUMSCRIBING CIRCLE DIAMETER    1.63  CENTER AT X =  Ø.749, Y =  Ø.373


 PRESS SYSTEM NO. 4, CAPACITY (TONS):  7ØØ., CONTAINER DIAMETER:  3.ØØØ
 BILLET -- DIAM:  2.875,  MAX LENGTH:  6.Ø, AREA:   6.5,  WEIGHT:   3.9
 BUTT LENGTH:  1.Ø,  RUN-OUT LENGTH (FT):  2Ø,  MAX. NO. OF OPENINGS:  1
 SPEED(FPM):   5., CYCLE TIME (SECS): 6Ø.,  EXTRUSION RATIO:  14.7


 WITH STANDARD BILLET, 1 HOLES, AND    5 FOOT EXTRUDED LENGTH,
 RECOVERY RATIO IS 73.9%, EXTRUSION RATIO:  14.7
 THIS GIVES  5 -  1 FOOT MULTIPLES, AND  Ø FOOT LOSS PER HOLE.


 WITH   5. FPM EXTRUSION SPEED, AND  6Ø. SEC. CYCLE TIME,
  3Ø.Ø PRESS CYCLES PER HOUR CAN BE RUN
 THIS GIVES   117. BILLET POUNDS AND    86. EXTRUDED POUNDS PER HOUR


 BEST YIELD WITH SPECIFIED OPENINGS
 NUMBER OF HOLES:  1, EXTRUDED LENGTH:   5
 BILLET LENGTH:  5.5, RECOVERY RATIO: 8Ø.3%, EXTRUSION RATIO:  14.7
  5 -  1 FOOT PIECES PER HOLE


 WITH  1 OPENINGS, THE BREAKTHROUGH LOAD IS   253. TONS.


 TONGUE PRESSURES:   2772. 68818.      Ø.
 TONGUE STRESSES:    1457. 1Ø276.      Ø.

 TONGUE PRESSURES:  71587.      Ø.     Ø.
 TONGUE STRESSES:   37647.      Ø.     Ø.
```

FIGURE 23. RESULTS OF ALEXTR ANALYSIS FOR "T" SECTION

On the first machining pass on the electrode, an 0.250-inch diameter cutter was used, although an 0.375-inch diameter cutter had been specified when generating the tape. This resulted in the electrode being left 0.062 inches oversize all the way around. After the roughing pass was completed, the excess stock beyond the projection was removed. This was done by manual operation of the joy-stick position controller which is a feature of the BCL CNC machine control. The finish cut on the electrode projection was then made using an 0.250-inch diameter cutter of the finish pass tape.

When generating the tape to machine the bearings, the following specifications were used:

- Diameter of cutter: 0.250 inch
- Cutter offset from die opening: 0.062 inch
- Minimum bearing length: 0.187 inch.

The die was machined to overall size in a lathe. The bearings were then NC machined into the back of the die. In machining the bearings, a cutter with 5-degree taper per side and an 0.250-inch diameter tip was used. The tapered cutter provided both a more rigid cutting tool than a standard ball mill of the same size, and also provided additional clearance from the opening at the back of the die. After heat treating the H-13 die steel to R_c 42-46, the actual section opening was EDM'ed through the die from the front. A 1.75 thick die backer was made with a straight through opening by hand rather than NC machining. The backer was also made from H-13 tool steel and hardened to R_c 42-46.

Figure 24 shows the finished die and backer. Figures 25 and 26 show a model of the die. This was made from aluminum and machined in the same way as the steel die. It was sectioned as shown to permit close examination of the bearing contours. Figure 27 is a plot of the cutter path which generated the bearings. Figure 28 is a plot of the cutter path used to cut the electrode. Both of these plots were obtained using NCDATA (see Appendix IX).

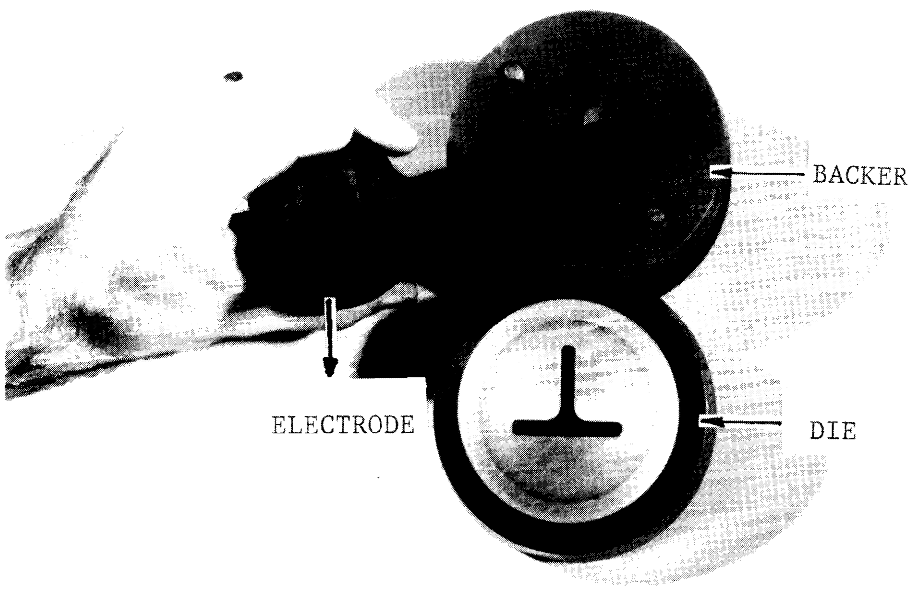


FIGURE 24. DIE AND BACKER FOR "T" SHAPE MADE USING ALEXTR AND EXTCAM

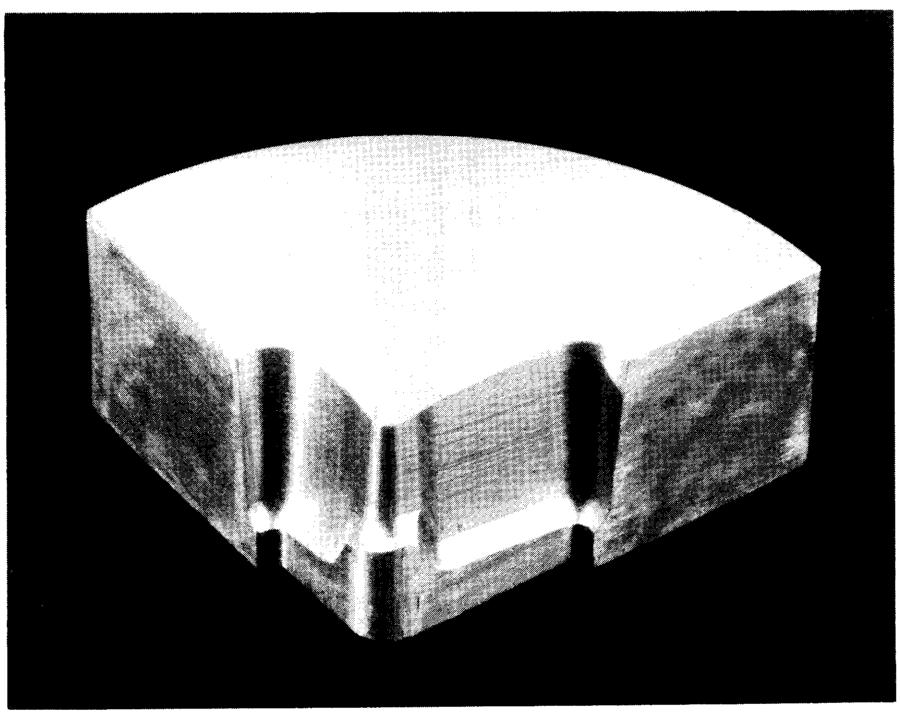


FIGURE 25. SECTIONED MODEL OF FLAT-FACE "T" DIE SHOWING VARIATION OF BEARINGS

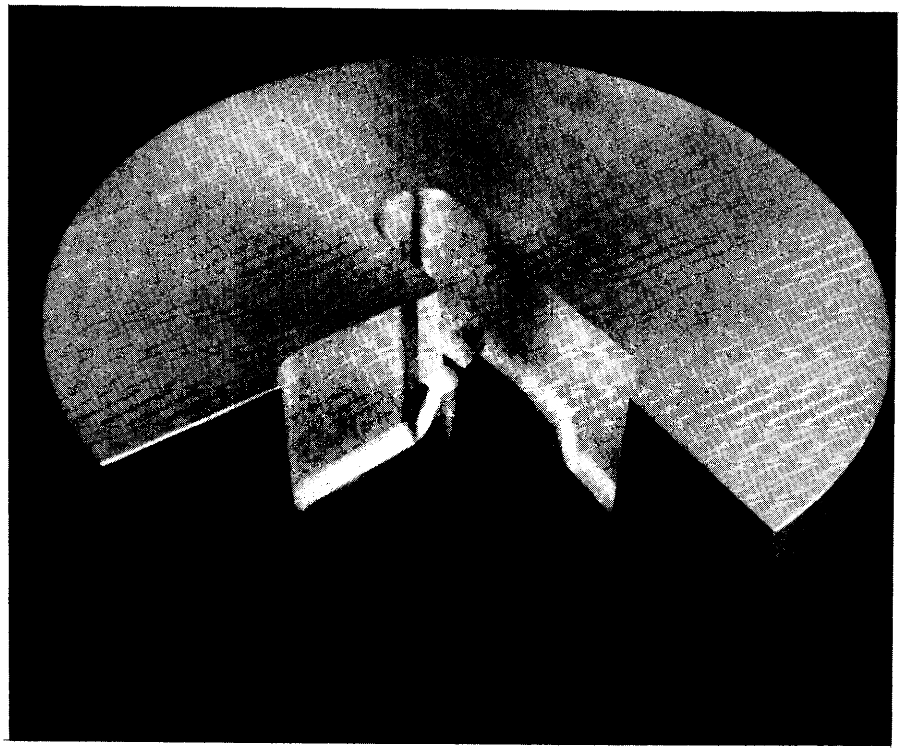


FIGURE 26. SECTIONED MODEL OF DIE SHOWING VARIATION OF BEARINGS

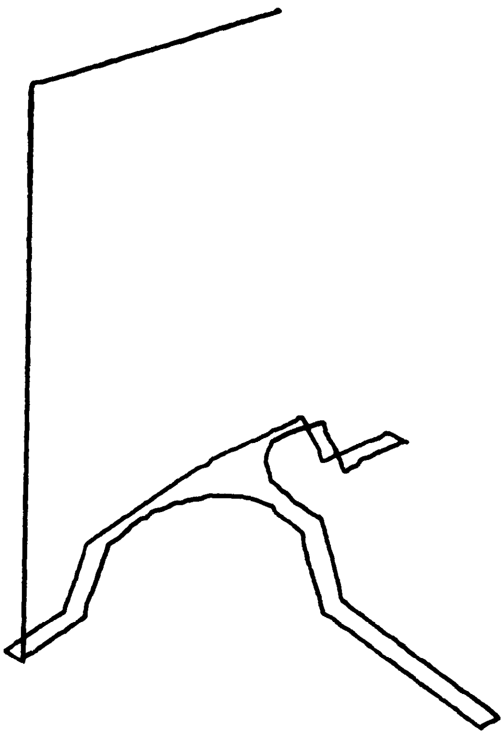


FIGURE 27. PLOT OF CUTTER PATH TO CUT BEARINGS INTO "T" DIE

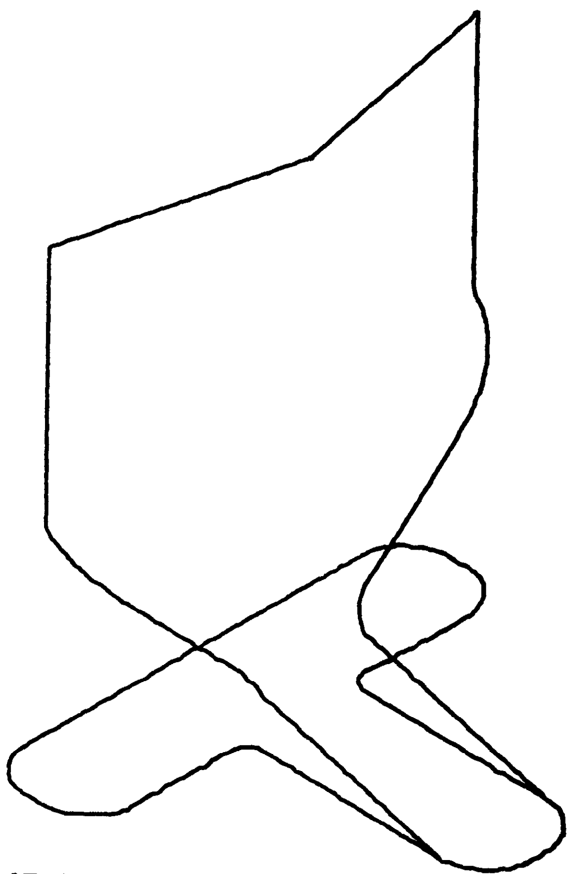


FIGURE 28. PLOT OF CUTTER PATH TO CUT EDM ELECTRODE FOR FRONT OPENING OF "T" DIE

SUMMARY AND RECOMMENDATIONS

Throughout the project, a number of extrusion engineers and designers were consulted as to the capabilities required in an extrusion die design system. The interaction with engineers and designers took place both at Battelle and various extrusion plants. The people consulted included those with some of the largest extrusion presses in the nation. The input provided by these individuals and companies was invaluable in the development of ALEXTR to its final form. This interaction also provided Battelle with an understanding of the designers thought processes, so the ALEXTR system would approach die design in a similar way. It is hoped that because of the latter consideration in the development of the system, ALEXTR will be more readily accepted by people who are not familiar with computers.

The ALEXTR and EXTCAM systems of programs were reviewed at BCL by a number of extruders after the programming was completed. Among those who viewed the system were individuals associated with companies which either had in use, or were developing their own extrusion CAD/CAM systems. The comments made concerning the system were generally favorable. It was felt the system considered all elements of die design normally encountered by a designer. The production analysis gave capabilities which are not now readily available and which could lead to more accurate price estimates. If a user added another subroutine which contained his unique cost factors, the manufacturing cost could be quickly and easily obtained. Furthermore, since the production analysis modules do not use the graphics other than to display the shape for user verification, these modules could be separated from the design portions of ALEXTR and implemented on a batch or time-share computer.

Specific suggestions for further work in the design sections of ALEXTR include the following:

- Permit cave (dish) compensation to be applied to multi-hole dies at the user's discretion. At present, it can be applied only to single-opening dies.
- In the bearing design module, the bearings should probably be shortened in the immediate area of an end of a section. This could be done by having the user indicate each end when specifying the bearing section thickness and transition points. It may also be possible for the ends to be located

by the program itself, although this would probably require verification by the user.

- When calculating the bridge stress for a multi-opening die, it should be possible for computer programs to identify the points defining the bridge area and inter-opening boundaries. An approach to the programs to do this has been conceived of, but has not been verified by additional study and testing.

Although not commented on by the extruders who critiqued the system, it is recognized by the authors that accurate prediction of extrusion load is possible only when correct flow stress data is used in the load calculation model. At present, the model uses data for flow stress which was obtained from published literature. The flow stress for certain alloys is known to be too high. Using data supplied by an extruder for maximum load, cylinder size, billet length, and section area, the model was run backwards in an attempt to determine the flow stress. Unfortunately, data concerning the shape perimeter and bearing length of the dies represented by the data was not available. Therefore, for these variables, values had to be assumed in the calculations. The results of using the supplied and assumed data gave flow stress values that were in the range of 40 to 50 percent of the values found in the literature. Although the flow stress data believed to be in error were revised downward, in line with the results for the analysis of the data supplied, the validity of the flow stress data is still open to question.

In conclusion, it is felt that although the ALEXTR and EXTCAM systems for extrusion production analysis and die design and manufacture are not perfect, they do offer the extruder or die manufacturer a valuable new tool to increase the productivity of his operation. The value of and need for such a system is attested to by the fact that at least two of the nations' largest extruders are using or developing similar systems and by the interest shown by extruders during the life of the project. As with any new technology, the degree to which it is accepted by the extrusion community will probably depend more on the individuals concerned than on the system itself. If a plant is open-minded concerning new technology, the ALEXTR can be of significant benefit, although certainly not without some teething problems. Extrusion CAD/CAM can, and will, be successfully applied to the degree that those responsible for its implementation want it to be successful.

APPENDIX I

EXTRUSION DESIGN AND ANALYSIS

APPENDIX I

EXTRUSION DESIGN AND ANALYSIS

Billet Yield Calculations

ALEXTR provides three ways in which the number of openings in a die may be determined. These are:

(1) By user specification

(2) Analytically based on maximum length

(3) Analytically based on maximum yield.

If the user specifies the number of openings, for instance in order to utilize existing back-up tooling, the percentage yield, number of cycles, and gross and net pounds per hour are calculated. These calculations are based on the default or specified parameters for the press, billet and extrusion. The optimum billet size to minimize loss is then determined and the above values are recalculated. The results of such a computation are shown in the first column of Table I-1. These calculations are all based on a volumetric balance. That is,

$$V_{billet} \geq V_{butt} + V_{extrusion} \tag{I-1}$$

The billet volume is found as:

$$V_{billet} = \frac{\pi}{4} D^2_{blt} L_{blt} \tag{I-2}$$

where D_{blt} = billet diameter

L_{blt} = billet length.

The butt volume (unextruded material remaining in the container at the end of the cycle) is found in a similar way, but using the container diameter and butt length.

The extruded volume is:

$$V_{ex} = (N_o)(N_m * L_m + 8) A_{ex} \tag{I-3}$$

TABLE I-1. RESULTS OF TYPICAL YIELD CALCULATIONS

Parameter		Method Specified: Operator	Maximum Length	Maximize Standard Billet Yield	Maximize Production Rate
Number of Holes		4	3	5	8
Standard:	Billet length	36	36	36	
	Yield	75.6	66.2	78.8	
	Billet lbs/hr	10607	9840	11504	
	Net lbs/hr	8020	6510	9060	
	Cycles/hr	39.5	36.6	42.8	
Optimum:	Billet length	31.6	27.5	33.3	33.9
	Yield	86.2	86.2	85.2	80.4
	Billet lbs/hr	9277	7521	10605	13021
	Net lbs/hr	8020	6510	9060	10468
	Cycles/hr	39.5	36.6	42.8	51.5
Extruded Length		104	120	88	56
Number of pieces/length		6	7	5	3

Press, Billet and Extrusion Parameters

Container Diameter	:	10	inches
Billet diameter	:	9.75	inches
Butt length	:	2	inches
Speed		135	FPM
Cycle time	:	45	sec
Run-out length	:	120	feet
Length multiple	:	16	feet
Loss per opening	:	8	feet

where N_o = number of openings in the die

N_m = integer number of pieces of length L_m extruded from each hole

L_m = finished length of each piece

A_{ex} = the cross-sectional area of the extrusion.

The quantity 8 in Equation I-3 represents the loss caused by the initial breakthrough of metal through the die, plus the material lost when gripping the extrusion to stretch it. A restriction on Equation I-3 is that the term $(N_m * L_m + 8)$ must be less than or equal to the runout length specified for the press. If the number of openings, N_o, is specified by the user, then the only unknown when combining I-1 and I-3 is N_m. This could be solved for by direct algebraic manipulation. To save on core memory space, however, a trial-and-error solution is used. This is because the same code is later used to find the best yield when both N_o and N_m are unknown.

Starting with $N_m = 1$, the volume of the extruded material is determined and compared to the amount available for extrusion. If the difference

$$d = V_{blt} - V_{ex} - V_{but} \tag{I-4}$$

is positive, N_m is incremented and V_{ex} recomputed. Once d becomes negative, N_m has been incremented too far and it is necessary to reduce it by 1.

The yield is calculated as the true, net pounds of extrusion resulting from the operation. That is,

$$Y_{\%} = (V_{blt} - V_{but} - 8\ N_o\ A_{ex})/V_{blt} \quad . \tag{I-5}$$

The optimum billet length for the specified number of openings is found by using the equality of Equation I-1. Knowing the volume of material to be extruded and using the minimum butt length specified, the minimum billet volume is found. Equation I-2 is then used to find the minimum billet length. Maximizing the yield does not affect the net pounds per hour value since the loss is minimized in the butt. Removing the butt is done after the extrusion cycle is completed and the time required is uneffected by the butt length.

If the user requests the number of openings to be based on maximizing the total extruded length, the following function is used:

$$L_R \geq N_m L_m + 8 \tag{I-6}$$

where L_R = specified runout length.

N_m is found as:

$$N_m = (L_R - 8)/L_m; \; N_m = 1,2,3, \ldots \tag{I-7}$$

and must be integer value. The value for N_m is then used with Equations I-2 and I-3 to determine N_o. N_o must also be an integer value.

The results of having the yield based on maximizing the extruded length is given in Column 2 of Table I-1. It is interesting to note that for the conditions specified in Table I-1, basing the number of holes on extruding the maximum length (three openings) gives a lower net pounds per hour value than the user-specified four opening die. The analysis assumes the speed of the extrusion leaving the die is constant and is not a function of the extrusion ratio. This analysis also does not consider the fact that more stretcher cycles will be required with the shorter extrusion from the four-opening die.

The third method for determining the number of openings in the die is to maximize the yield from the specified billet. In this case, both N_m and N_o are independent variables and a solution must be generated for each case. The yield is calculated for every value of N_m and N_o and the values which give the maximum yield are retained and reported. The results of this approach are given in Column 3 of Table I-1. The optimum billet length for this setting of N_m and N_o is also calculated. It is interesting to note that the conditions which optimize yield from a standard billet do not minimize the yield when the billet size is optimized. The percentage difference, however, in this case at least is small.

The maximum yield method also calculates the number of openings and length of each piece such that the pounds per hour value is maximized. Again, this is done by computing the net pounds per hour for all combinations of N_m and N_o. The results for this approach are given in the fourth Column of Table I-1. When the number of openings giving the best yield from a standard billet is different from the number of openings giving the highest production rate in pounds per hour, the values for these two conditions are printed. The user is then asked to select which value he wishes to use in subsequent steps.

The system is designed so that the user may repeat this entire process as often as he wishes. He could, in fact, repeat the "Operator" mode for every possible value of N_o, thus calculating the best yield from standard and optimum billets and the highest production rate for all conditions.

CALCULATION OF EXTRUSION LOAD

The upper-bound method is used to derive an expression for estimating the load required to extrude a billet through a multi-orifice flat-face die. The following assumptions are made in the analysis:

(1) The flow stress of the material, $\bar{\sigma}$, is constant throughout the deformation zone.

(2) No lubricant is used and, therefore, the friction shear stress at the billet container interface is equal to the flow stress of the material in shear $(\sigma/\sqrt{3})$.

(3) The component of pressure due to plastic deformation in extruding a nonsymmetric section (p_s) is related to that for a circular section of equal cross-sectional area (p_c) as follows:

$$p_s = p_c \; (1.2 - 0.2 \; \ell_c/\ell_s) \quad ,$$

where ℓ_s = perimeter of the nonsymmetric section

ℓ_c = perimeter of a circular section of equal cross-sectional area.

Thus, the ratio (p_s/p_c) lies between 1.2 and 1.0.

Extrusion of a Circular Section through a Single Hole Flat-Face Die

The total rate of energy supplied ($\dot{E}_t$) at any instance must be equal to the sum of the rates of energy dissipated during the extrusion process. This is expressed by the equation:

$$\dot{E}_t = \dot{E}_{fc} + \dot{E}_i + \dot{E}_{sh} + \dot{E}_{fd} + \dot{E}_{f\ell} \quad . \tag{I-8}$$

The symbols used are explained in the Nomenclature. The various terms of Equation (I-8) are given by[1,2*]:

$$\dot{E}_t = P_t \cdot V_o = A\, p_{avg}\, V_o \tag{I-9}$$

$$\dot{E}_{fc} = \frac{1}{\sqrt{3}} \pi D \, (L - L_1) \cdot \bar{\sigma}\, V_o \tag{I-10}$$

$$\dot{E}_i = A\, \bar{\sigma}\, f(\alpha) \cdot \ln\left(\frac{A}{A_f}\right) V_o \tag{I-11}$$

$$\dot{E}_{sh} = \frac{2}{\sqrt{3}} A\, \bar{\sigma} \left[\frac{\alpha}{\sin^2\alpha} - \cot\alpha\right] V_o \tag{I-12}$$

$$\dot{E}_{fd} = \frac{1}{\sqrt{3}} A\, \bar{\sigma}\, (\cot\alpha) \cdot \ln\left(\frac{A}{A_f}\right) V_o \tag{I-13}$$

$$\dot{E}_{f\ell} = \frac{1}{\sqrt{3}} \ell_c\, L_d\, \bar{\sigma}\, V_f \tag{I-14}$$

where

$$f(\alpha) = \frac{1}{\sin^2\alpha}\left[1 - (\cos\alpha)\sqrt{1 - \frac{11}{12}\sin^2\alpha} + \frac{1}{\sqrt{11 \times 12}} \ln \frac{1 + \frac{11}{12}}{\sqrt{\frac{11}{12}}\cos\alpha + \sqrt{1 - \frac{11}{12}\sin^2\alpha}}\right] \tag{I-15}$$

---

* References are listed at the end of the appropriate Appendix.

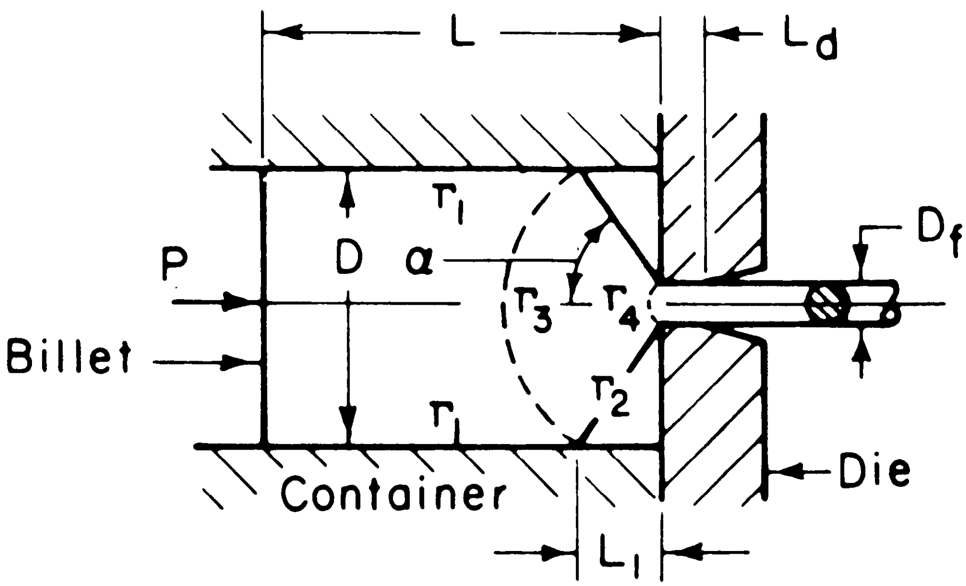


FIGURE I-1. EXTRUSION THROUGH A SINGLE HOLE FLAT-FACE DIE

Substituting Equations (I-9) through (I-13) into equation (I-8)

$$\frac{P}{\bar{\sigma}} = \frac{1}{\sqrt{3}} \pi D \left\{ L - \left(\frac{D-D_f}{2}\right) \cot\alpha \right\} + A \left\{ F(\alpha) \cdot \ln\left(\frac{A}{A_f}\right) \right. \tag{I-16}$$

$$\left. + \frac{2}{\sqrt{3}} \left(\frac{\alpha}{\sin^2\alpha} - \cot\alpha\right) + \frac{1}{\sqrt{3}} \cot\alpha \ln\left(\frac{A}{A_f}\right) + \frac{1}{\sqrt{3}} \frac{\ell_c L_d}{A_f} \right\} .$$

The dead zone cone angle, α, is an unknown parameter. Using the upper-bound theorem, α can be determined from the condition:

$$\frac{\partial P}{\partial \alpha} = 0 \quad . \tag{I-17}$$

Neglecting $(\partial L_1/\partial\alpha)$ in Equation (I-17), the values of α for various reductions have been derived[2]. α is given approximately by the following relationships:

$$\begin{array}{lll} \text{For} & 0 < r \leq 5 & \alpha = 3r \\ & 5 < r \leq 80 & \alpha = 15.0 + 0.56\ (r - 5.0) \\ & 80 < r < 100 & \alpha = 57.0 + 0.9\ (r - 80.0) \quad , \end{array} \tag{I-18}$$

where

$$r = \left[1 - \frac{A}{A_f} \right] \times 100 \quad . \tag{I-19}$$

Extrusion of a Nonsymmetric Section through a Single Hole Flat-Face Die

Using the assumption stated earlier, the Equation (I-16) is modified to give the following expression for extrusion load:

$$\frac{P}{\bar{\sigma}} = \frac{1}{\sqrt{3}} \pi D \left[L - \left(\frac{D - D_e}{2}\right) \cot\alpha\right] + A \left[f(\alpha) \ln\left(\frac{A}{A_f}\right)\right.$$

$$\left(1.2 - 0.2 \frac{\ell_c}{\ell_s}\right) + \frac{2}{\sqrt{3}} \left(\frac{\alpha}{\sin^2\alpha} - \cot\alpha\right) + \frac{1}{\sqrt{3}} \cot\alpha \left(\frac{A}{A_f}\right)$$

$$\left. + \frac{1}{\sqrt{3}} \ell_s \frac{L_d}{A_f}\right] , \qquad \text{(I-20)}$$

where $D_e = \sqrt{\dfrac{4A_f}{\pi}}$. (I-21)

<u>Extrusion of Nonsymmetric Section through Multi-Hole Flat-Face Die</u>

Here, the assumption is made that in extrusion through an N-hole flat-face die, the billet is essentially divided into N-billets at the flat face of the die. Based on this assumption, Equation (I-20) is modified to the following expression for load required to extrude through a multi-hole flat-face die:

$$\frac{P}{\bar{\sigma}} = \frac{1}{\sqrt{3}} \pi D \left[L - \left(\frac{D_b - D_e}{2}\right) \cot\alpha\right] + A \left[f(\alpha) \ln\left(\frac{A}{NA_f}\right)\right.$$

$$\left(1.2 - 0.2 \frac{\ell_c}{\ell_s}\right) + \frac{2}{\sqrt{3}} \left(\frac{\alpha}{\sin^2\alpha} - \cot\alpha\right) + \frac{1}{\sqrt{3}} \cot\alpha \ln\left(\frac{A}{NA_f}\right)$$

$$\left. + \frac{1}{\sqrt{3}} \ell_s \frac{L_d}{A_f}\right] , \qquad \text{(I-22)}$$

$$D_e = \sqrt{\frac{4A_f}{\pi}} , \quad D_b = \sqrt{\frac{4A}{N\pi}} . \qquad \text{(I-23)}$$

The angle α is given by:

For $0 < r_e \le 5$	$\alpha = 3\, r_e$
$5 < r_e \le 80$	$\alpha = 15.0 + 0.56\ (r_e - 5.0)$
$80 < r_e < 100$	$\alpha = 57.0 + 0.9\ (r_e - 80.0)$

where $$r_e = \left[1 - \frac{A}{NA_f}\right] \times 100 \ .$$

NOMENCLATURE

A = cross-sectional area of the billet

A_f = final area of the extrusion (product)

D = diameter of the billet

D_b = effective diameter of the portion of billet feeding one hole (Equation I-23)

D_e = effective diameter of the extrusion as defined by Equation (I-23)

D_f = diameter of the extrusion

$\dot{E}_{fc}$ = rate of energy dissipated due to friction at the container wall

$\dot{E}_{fd}$ = rate of energy dissipated due to friction at the die surface

$\dot{E}_{f\ell}$ = rate of energy dissipated due to friction at the die land

$\dot{E}_i$ = rate of energy dissipated due to plastic deformation

$\dot{E}_{sh}$ = rate of energy dissipated due to shearing caused by tangential velocity discontinuities

$\dot{E}_t$ = total rate of energy supplied

L = length of the billet in contact with the container

L_d = length of the die land

N = number of holes

P_{avg} = mean extrusion pressure

P_t = total extrusion load

V_o = speed of the ram

V_f = speed of the extrusion **(product)**

α = angle that the die surface makes with the extrusion axis

$\bar{\sigma}$ = average flow stress

CORRECTION OF DIE-HOLE DIMENSIONS FOR THERMAL EFFECTS

To obtain the product with the desired dimensions, the die-hole dimensions have to be corrected for two thermal effects.

(a) Die Expansion. Preheating of the dies and heating during the extrusion change the dimensions of the die holes from the manufactured dimensions at the ambient temperature.

(b) Extrusion Shrinkage. The hot extrusion shrinks upon cooling to room temperature. The dimensions of the extrusion at ambient temperature are thus different from the dimensions of the die hole from which it is extruded.

The change, $\Delta \ell$, in any linear dimension ℓ of material due to a temperature change from T_1 to T_2 is given by:

$$\Delta \ell = \ell \int_{T_1}^{T_2} \alpha \, dt \quad , \qquad \text{(I-24)}$$

where α is the coefficient of thermal expansion of the material. The percent change is given by:

$$S = \frac{\Delta \ell}{\ell} \times 100 \quad . \qquad \text{(I-25)}$$

The average coefficient of thermal expansion for die steel H13 is 6.88×10^{-6} in/in/F in the temperature range of 100-800 F, whereas for most aluminum alloys, α is 13×10^{-6} to 15×10^{-6} in/in/F in 68-275 F temperature range.

Die Expansion

An accurate analytical prediction of change in dimensions of the die holes due to the change in die temperature is very difficult. Fortunately, as shown below by means of an approximate analysis, the dimensional changes due to die expansion are expected to be extremely small and can be safely neglected.

Figure I-2 shows an N-hole die for extruding circular sections (rods). Due to symmetry, it is sufficient to consider the expansion of one sector OAB of the die. Let

D_1 = diameter of the die blank at ambient temperature T_1

r_1 = radius at which the CG (or the center) of the circle is located at the ambient temperature (Figure I-2)

d_1 = diameter of the circular hole at the ambient temperature.

When the die temperature is raised from ambient to T_2, the die blank expands. Without any holes, the thickness as well as the diameter of the die hole will increase, as seen in Figure I-3a. Due to axial symmetry, any point on the plane OAB will move radially outwards by a distance given by Equation (I-24) with ℓ as the distance of the point from the center O. As shown below, an imaginary circle C_1 with center p will take the shape of a bigger circle C_2 with center p'. Due to temperature change, p will move to p' and q to q' such that

$$\frac{Op'}{Op} = \frac{Oq'}{Oq} = 1 + \int_{T_1}^{T_2} \alpha \, dt \quad . \tag{I-26}$$

Therefore, triangles Opq and Op'q' are similar and

$$\frac{Op'}{Op} = \frac{p'q'}{pq} = 1 + \int_{T_1}^{T_2} \alpha \, dt \quad . \tag{I-27}$$

Thus, the locus of point q' corresponding to the points on circle C_1 will be a circle with radius p'q'.

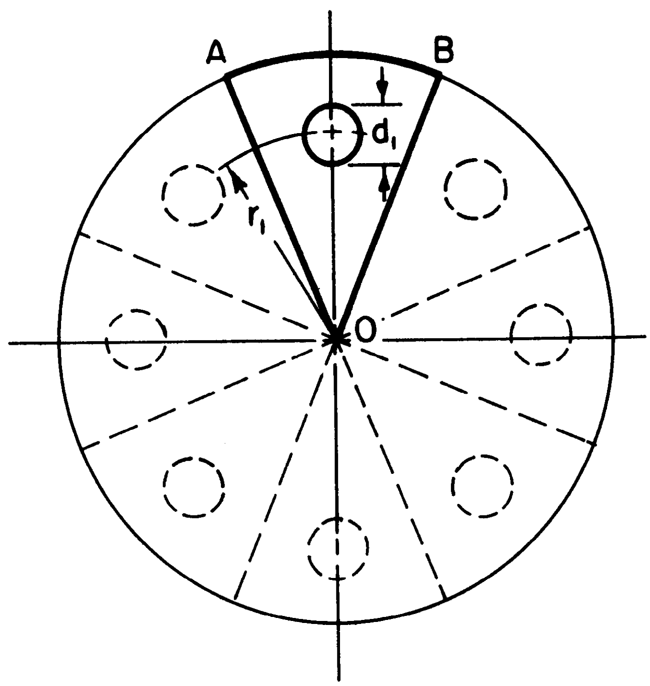
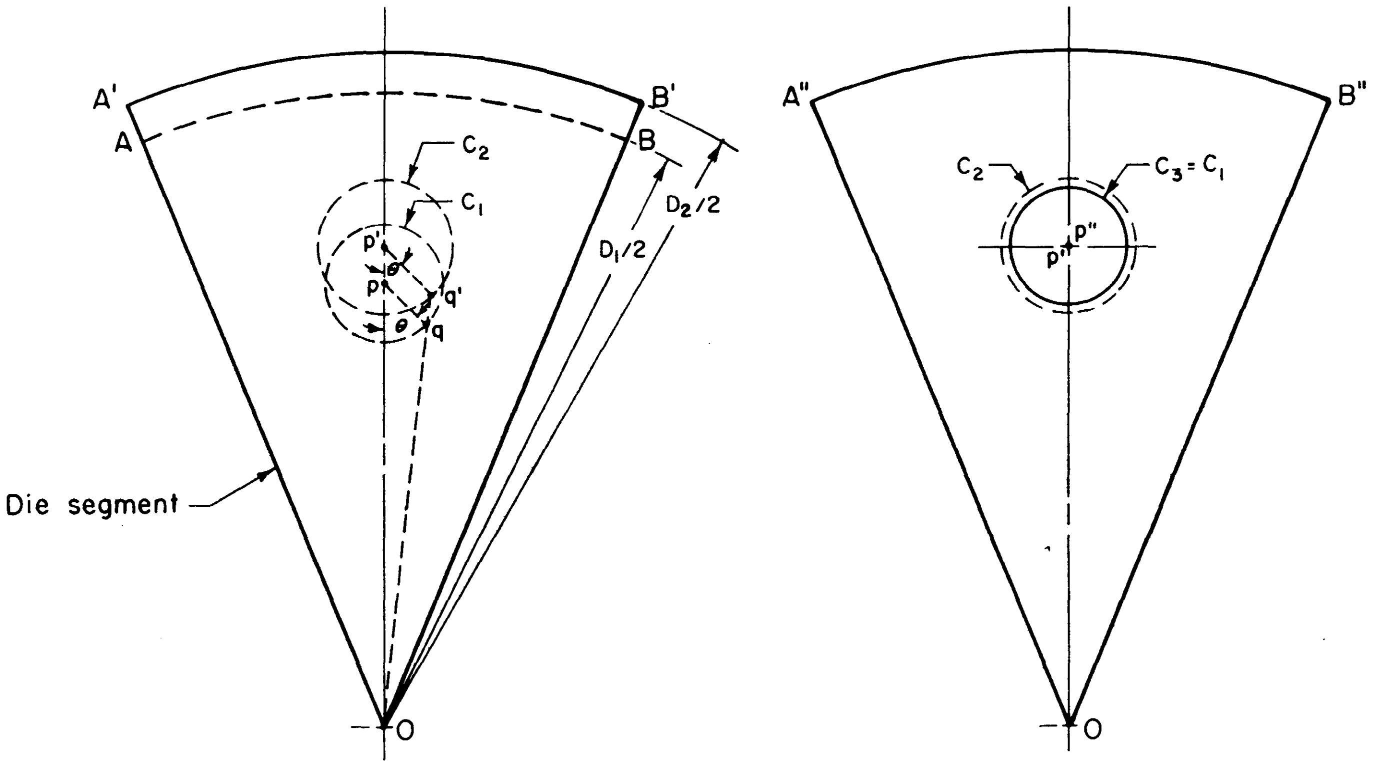


FIGURE I-2. N-HOLE DIE FOR EXTRUDING CIRCULAR ROD



(a) Expansion of a Die Blank With no Holes

(b) Expansion of a Die Segment with One Circular Hole

FIGURE I-3. CHANGE IN HOLE-DIMENSIONS WITH CHANGE IN DIE TEMPERATURE

C_1 will expand to C_2 when the die segment does not have any hole. For die segment with the hole, there is no material in the hole to expand. Therefore, the expansion of the circular disc C_1 must be subtracted from C_2 to obtain the configuration of the hole after temperature change. With q'' as a point on the circumference of the expanded hole, this gives

$$p'q'' = p'q' - \left[\int_{T_1}^{T_2} \alpha \, dT \right] pq = pq \quad . \tag{I-28}$$

Thus, increase in die temperature does not affect the shape and size of the circular hole. It merely shifts its center according to the relation (I-26). In extrusion of aluminum alloys, T_2 is approximately 900 F which gives percentage shift, S, in center as

$$S = \frac{OP' - OP}{OP} \times 100 = 6.88 \times 10^{-6} (900 - 80) \times 100 = 0.56 \text{ percent} .$$

This small shift will not affect the design process and can be neglected.

With noncircular holes, due to nonsymmetry, there will be some distortion of the aperature shape. However, since it will be some small percentage of the total dimensional change which is less than one percent, it can be safely neglected.

Extrusion Shrinkage

The correction for thermal shrinkage of the extrusion can be obtained by using Equation (I-26). The coefficient of expansion α for aluminum alloys is, however, dependent upon the temperature. Wiley[3] developed the following relation for calculating linear expansion over 0 to 1000 F temperature range.

$$L_t = L_o \left[1 + C \left(12.19t + .003115 \, t^2\right) 10^{-6}\right] \quad , \tag{I-29}$$

where L_o = length at temperature, 0 F

L_t = length at temperature, t F

C = alloy constant given in Table I-2 for various aluminum alloys[4].

Using relation (I-29), the corrected orifice dimension ℓ_f, which will give the desired section dimension ℓ_o at ambient temperature T_a is:

$$\ell_f = \ell_o \left(\frac{1 + C\ (12.19\ T_e + .003115\ T_e^{\ 2})10^{-6}}{1 + C\ (12.19\ T_a + .003115\ T_a^{\ 2})10^{-6}} \right) , \qquad \text{(I-30)}$$

where T_e is the extrusion temperature or more correctly the temperature of the extruding product at the die exit. Table I-2 gives the range of temperature at which various aluminum alloys are extruded.

As an example, consider the extrusion of Aℓ 6061 at temperature 850 F. From Equation (I-30)

$$\ell_f = 1.0115\ \ell_o \quad .$$

Thus, the orifice (hole) dimensions are obtained by increasing the metal dimensions by 1.15 percent. This agrees with the conventional practice.

TECHNIQUE FOR CORRECTING HOLE DIMENSIONS TO INCLUDE THERMAL SHRINKAGE

All the metal dimensions have to be increased, according to Equation (I-30). It can be easily proved that this is accomplished by expanding each point on the section perimeter radially away from a single point (pole) according to Equation (I-30). A convenient pole is the CG of the section. As shown in Figure I-4 for a U shape, the corrected profile is obtained by expanding such that

$$\frac{OA'}{OA} = \frac{OB'}{OB} = \frac{OC'}{OC} = \frac{OD'}{OD} \cdot \cdot \cdot \cdot \cdot = \frac{1 + C\ (12.19\ T_e + .003115\ T_e^{\ 2})10^{-6}}{1 + C\ (12.19\ T_a + .003115\ T_a^{\ 2})10^{-6}}$$

From similar triangles OAB and OA'B' and so on ,

$$\frac{A'B'}{AB} = \frac{B'D'}{BD} \cdot \cdot \cdot \cdot \cdot = \frac{1 + C\ (12.19\ T_e + .003115\ T_e^{\ 2})10^{-6}}{1 + C\ (12.19\ T_a + .003115\ T_a^{\ 2})10^{-6}}$$

Thus, by this technique, the metal dimensions are increased as desired.

TABLE I-2. ALLOY CONSTANT, C, AND EXTRUSION TEMPERATURE RANGE FOR ALUMINUM ALLOYS

ALLOY	ALLOY CONSTANT, C[1-3]	EXTRUSION TEMPERATURE
EC	1.000	750-950
1060	1.000	850-950
1100	1.000	800-950
2017	.970	-
2024	.970	600-860
3003	.985	850-1000
5056	1.025	-
5154	1.015	-
6053	.980	-
6061	.990	600-850
6063	.990	750-850
7075	.990	550-850

NOTES:
1. Constants are applicable to alloys in annealed tempers.
2. With heat-treatable alloys, the use of Equation (I-29) and alloy constants is restricted to temperatures below 600 F.
3. With wrought alloys 7075, application of the equation and alloy constants is restricted to temperatures below 400 F.

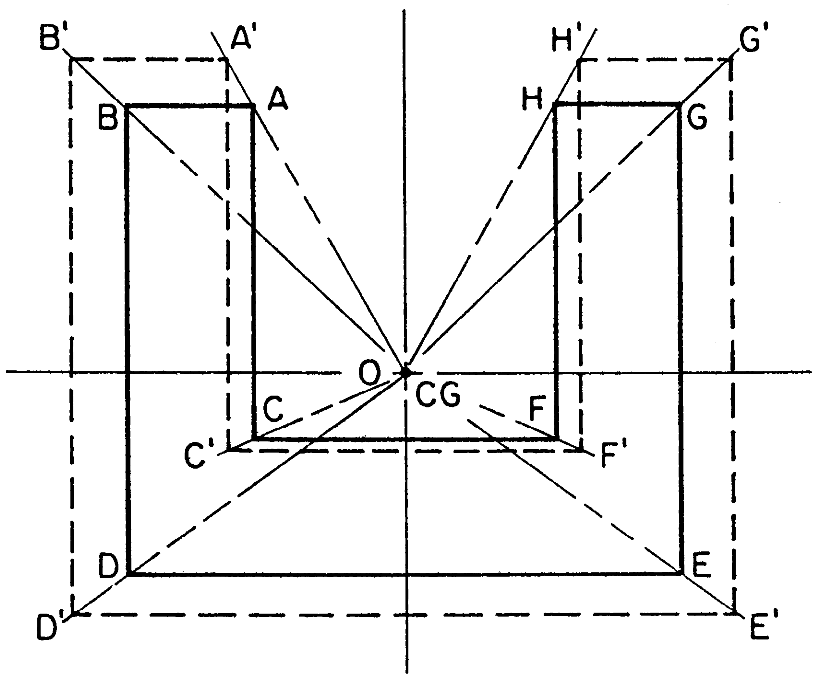


FIGURE I-4. TECHNIQUE FOR CORRECTING DIE-HOLE PROFILE FOR THERMAL SHRINKAGE

TOOLING DESIGN

During extrusion, the deforming metal exerts pressure on the die which is supported by the die and the back-up tooling. The back-up tooling usually consists of a backer, a bolster and sometimes an additional sub-bolster. These tooling members must be designed to withstand the extrusion pressures.

The tooling design involves (a) selection of proper tooling materials, (b) determination of thicknesses of various members, and (c) need for backer and/or bolster support. In the CAD/CAM computer system for the flat-face dies, special computer programs are included which allow the designer to perform these tooling design operations. This appendix gives the mathematical basis used in these computer programs.

Due to the metal pressure, the support tooling can fail by (a) shearing through the thickness and (b) plastic (permanent) deformation of tongues by bending The stresses in the tooling for given thicknesses of the tooling members can be determined as discussed below. The following assumptions are made in calculating stresses:

(a) Uniform pressure equal to the average extrusion pressure acts on the die surface.

(b) There is no friction at the contacting surfaces of the tooling members. The members slide against each other freely.

Tool Strength Against Shearing

The average shear stress, τ, in the areas between the die openings is given by:

$$\tau = \frac{pA}{A_1 + A_2 + A_3} , \qquad \text{(I-31)}$$

where p = average extrusion pressure

A = plan area of the die face between the die openings

A_1, A_2, A_3 = shear-resisting cross-sectional area (perpendicular to the die face) of the die, backer and bolster, respectively.

As an example, consider a 4-hole die for L sections (Figure I-5). The average shearing stress acting on the tooling along AB, DE, GH, JK planes is:

$$\tau = \frac{p \cdot \text{Area ABC...JKA}}{(AB+DE+GH+JK)t_1 + (AB+DE+GH+JK-8C_1)t_2 + (AB+DE+GH+JK-8C_1-8C_2)t_3} \tag{I-32}$$

where C_1 = clearance between die and backer openings

C_2 = clearance between backer and bolster openings

t_1, t_2, t_3 = thickness of die, backer and bolster, respectively.

Bending of Tongues in the Extrusion Tooling Stack

The tongue is taken as a cantilever beam uniformly loaded by a pressure equal to the average extrusion pressure, p. The basic relations for deflection, y, and maximum resultant stress, σ_v, at the cantilever base, are given by:

$$y = \frac{p}{24EI}(x^4 + 6\ell^2x^2 - 4\ell x^3) \tag{I-33}$$

where p = extrusion pressure

E = Young's modulus of elasticity (~ 27×10^6 psi at 800 F for H13 die steel)

I = moment of inertia. I is equal to $(t^3/12)$ for a rectangular section of unit width and thickness, t.

ℓ = length of the beam

x = location where deflection is measured. x is measured from the fixed end,

and

$$\sigma_v = \sqrt{\sigma^2 + 3\tau^2} \tag{I-34}$$

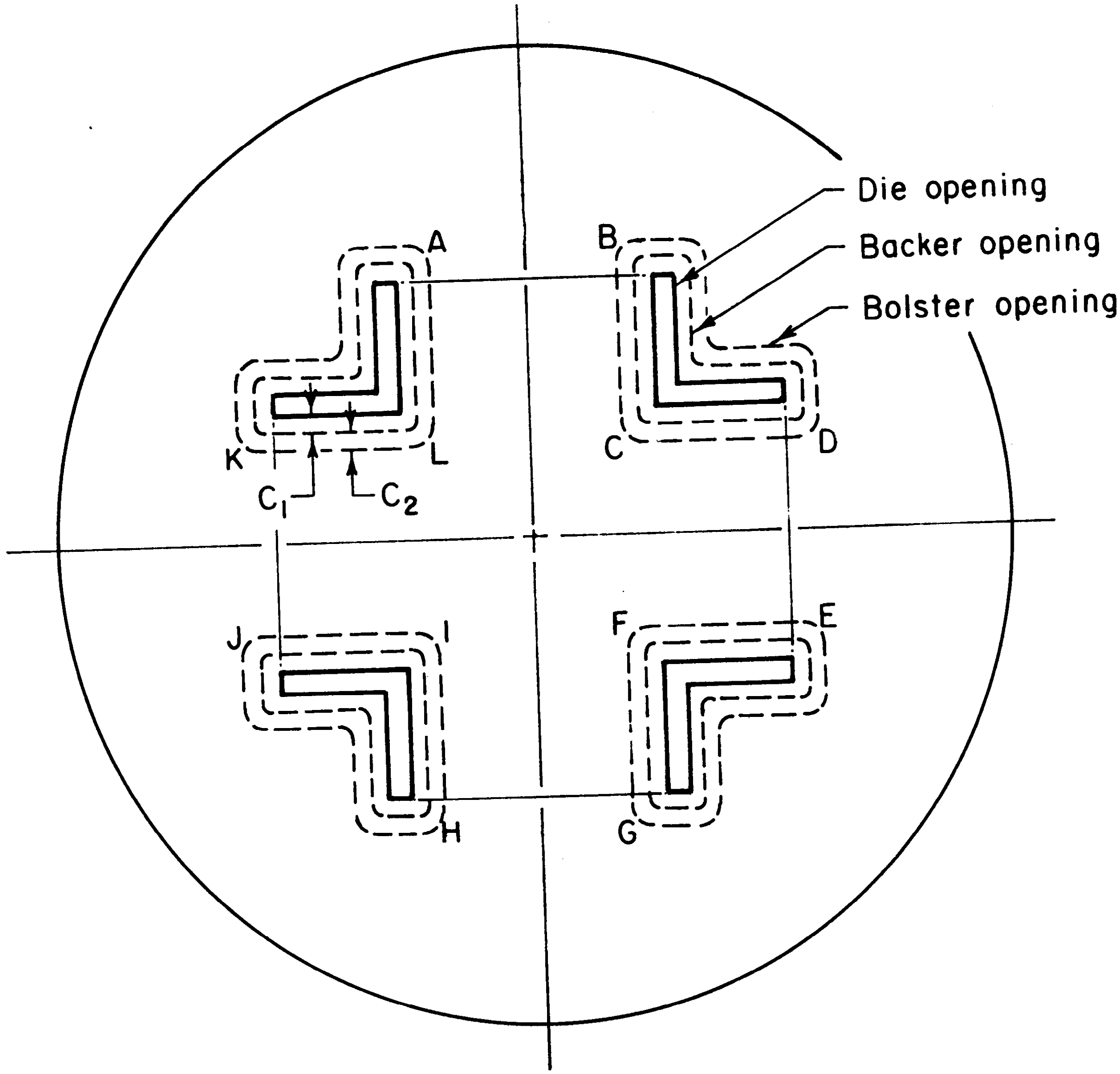


FIGURE I-5. TOOL STRENGTH AGAINST SHEARING (Points A,B. . .K refer to the corners of the die openings) with $\sigma = 3p\ell^2/t^2$ for a beam with rectangular cross section

and $\tau = p\ell/t$

σ is the stress caused by bending and τ is the shear stress due to the uniformly distributed load.

In the extrusion tool stack, backer is placed behind the die and the bolster is placed behind the backer. The stack will act as one thick plate if there is no relative movement at the die-backer and backer-bolster interfaces. On the other hand, the stack will act as separate plates if there is free relative movement at the interfaces. The deflection and bending stresses will be different for the two cases. In actual practice, the friction conditions, depending upon the surface finish and surface cleanliness, will be in between the two extreme cases. The same will be true for the deflection and bending stresses. Here, the stresses are calculated assuming free movement at the contacting surfaces.

In calculating stresses, three cases are considered. In Case I, the members supporting the load are die, backer and bolster. In Case II, the supporting members are die and backer only, and in Case III, die is the only member supporting the load.

<u>Case I - Die, Backer and Bolster</u>

Considering tongue as a cantilever beam, the deflection of each component of tool stack is determined by the net pressure acting on that component. Net pressure is defined as the pressure on the front face, minus the pressure on the back face. The net pressures in the die tongue, backer tongue and bolster tongue are determined from the condition that deflection of all tools at the free end of the bolster is equal. From this condition, one obtains:

$$y_3 = \frac{p_1}{24EI_1}(\ell_3^4 + 6\ell_1^2\,\ell_3^2 - 4\ell_1\ell_3^3) = \frac{p_2}{24EI_2}(\ell_3^4 + 6\ell_2^2\,\ell_3^2 - 4\ell_2\ell_3^3)$$

$$= \frac{p_3}{24EI_3}\cdot 3\ell_3^4 \quad , \qquad \text{(I-35)}$$

where subscripts 1, 2 and 3 refer to the die, backer and bolster as shown in Figure I-6.

Also $$p_1 + p_2 + p_3 = p \quad . \tag{I-36}$$

From Equations (I-35) and (I-36)

$$p_3 = p\left[\frac{1}{1 + \frac{I_1}{I_3}\left(\frac{3\ell_3^{\,4}}{\ell_3^{\,4} + 6\ell_1^{\,2}\,\ell_3^{\,2} - 4\ell_1\ell_3^{\,3}}\right) + \frac{I_2}{I_3}\left(\frac{3\ell_3^{\,4}}{\ell_3^{\,4} + 6\ell_2^{\,2}\,\ell_3^{\,2} - 4\ell_2 p_3^{\,3}}\right)}\right]$$

$$p_1 = p_3 \quad \frac{I_1}{I_3}\left[\frac{3\ell_3^{\,4}}{\ell_3^{\,4} + 6\ell_1^{\,2}\,\ell_3^{\,2} - 4\ell_1\ell_3^{\,3}}\right]$$

$$p_2 = p_3 \quad \frac{I_2}{I_3}\left[\frac{3\ell_3^{\,4}}{\ell_3^{\,4} + 6\ell_2^{\,2}\,\ell_3^{\,2} - 4\ell_2\ell_3^{\,3}}\right] \tag{I-37}$$

Case II - Die Plus Backer

Using the condition that the deflections of the die and the backer are equal at the free end of the backer, one obtains:

$$y_2 = \frac{p_1}{24EI_1}\left(\ell_2^{\,4} + 6\ell_1^{\,2}\,\ell_2^{\,2} - 4\ell_1\ell_2^{\,3}\right)$$

$$= \frac{p_2}{24EI_2}\left(\ell_2^{\,4} + 6\ell_2^{\,4} - 4\ell_2^{\,4}\right) = \frac{3\ell_2^{\,4}}{24EI_2} \cdot p_2 \tag{I-38}$$

Also, $$p_1 + p_2 = p \tag{I-39}$$

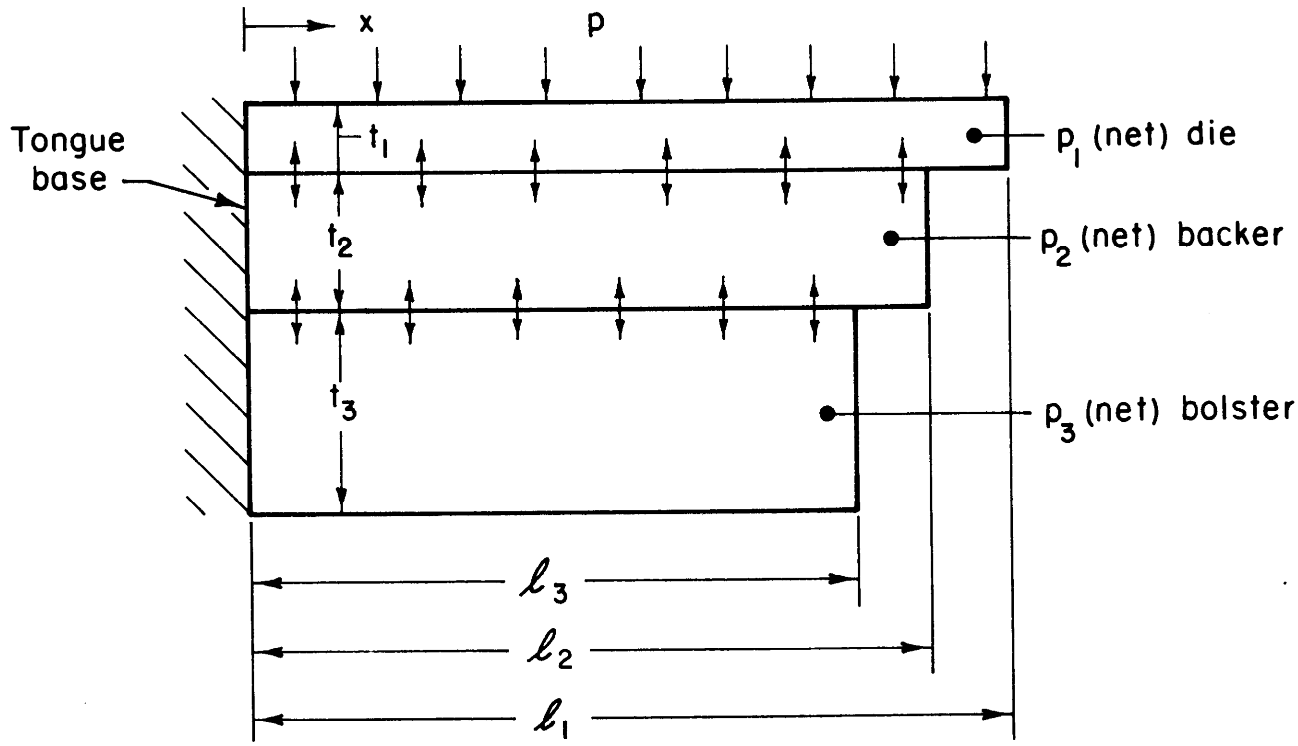


FIGURE I-6. NET PRESSURES ON THE COMPONENTS OF THE TOOL STACK AT A TONGUE which gives

$$p_2 = p\left[\frac{1}{1+\frac{I_1}{I_2}\left(\frac{3\ell_2^4}{\ell_2^4+6\ell_1^2\ell_2^2-4\ell_1\ell_2^3}\right)}\right]$$

$$p_1 = p_2\,\frac{I_1}{I_2}\left[\frac{3\ell_2^4}{\ell_2^4+6\ell_1^2\ell_2^2-4\ell_1\ell_2^2}\right] \quad . \tag{I-40}$$

<u>Case III - Die Only</u>

In this case,

$$p_1 = p \quad . \tag{I-41}$$

The maximum resultant stresses acting at the bases of the die tongue, the backer tongue and the bolster tongue are:

$$\sigma_{v_1} = \sqrt{\left(\frac{3p_1\ell_1^2}{t_1^2}\right)^2 + 3\left(\frac{p_1\ell_1}{t_1}\right)^2}$$

$$\sigma_{v_2} = \sqrt{\left(\frac{3p_2\ell_2^2}{t_2^2}\right)^2 + 3\left(\frac{p_2\ell_2}{t_2}\right)^2} \tag{I-42}$$

$$\sigma_{v_3} = \sqrt{\left(\frac{3p_3\ell_3^2}{t_3^2}\right)^2 + 3\left(\frac{p_3\ell_3}{t_3}\right)^2}$$

REFERENCES

(1) Nagpal, V., and Altan, T., "Computer-Aided Design and Manufacturing for Extrusion of Aluminum, Titanium and Steel Structural Parts (Phase I)", Army Materials and Research Center (AMMRC) CTR 76-6, (1976).

(2) Avitzur, B., "Metal Forming: Processes and Analysis", McGraw-Hill Book Company, New York, (1968), 153-170.

(3) Wiley, L. A., ALCOA Research Laboratories (1957).

(4) ALCOA Aluminum Handbook, Aluminum Company of America, Pittsburgh, Pennsylvania, 1962.

APPENDIX II

DESCRIPTION OF GENERAL PURPOSE COMPUTER PROGRAMS

APPENDIX II

DESCRIPTION OF GENERAL PURPOSE COMPUTER PROGRAMS

Determination of Circumscribing Circle Diameter

Given a set of X, Y coordinate points, a circle can be found such that it is the smallest among all the circles which encompass all the points. Such a circle is found by first establishing the center of a trial circle and then testing all points relative to this center. If any point is further away from the center than the radius of the circle, a new trial circle must be determined.

Given three points, either a two or a three point circle can be found to circumscribe all the points. The number of points defining the circle depends on the nature of the triangle formed by the points. For obtuse (1 angle greater than 90 degrees) or right triangles, the circumscribing circle (CC) is defined by the two end points of the largest line of the triangle as shown in Figure II-1a and II-1b. For acute triangles (no angle greater than 90), the CC is defined by all three points of the triangle, Figure II-1c. It should be noted that a three point circle can be fitted to an obtuse triangle. However, such a circle will be larger than the two-point circle for the same set of points.

Thus, to find the CC for a set of three points, it is first necessary to determine if the triangle formed by the three points is acute or obtuse. Referring to Figure II-2, the lengths of each side are found as:

$$ASQ = [X(I1) - X(I2)]^2 + [Y(I1) - Y(I2)]^2 \tag{II-1}$$

$$BSQ = [X(I1) - X(I3)]^2 + [Y(I1) - Y(I2)]^2 \tag{II-2}$$

$$CSQ = [X(I2) - X(I3)]^2 + [Y(I1) - Y(I2)]^2 \tag{II-3}$$

The longest side of the triangle is found (ASQ in this case), and then this is compared to the other two sides.

(*) A portion of this Appendix was originally prepared for the Army Program on "Computer-Aided Design and Manufacturing for Closed Die Forging of Track Shoes and Links, AMMRC No. CTR 76-21, prepared by Battelle's Columbus Laboratories under Contract No. DAAG-46-75-C-0041, July, 1976. It is included here for the sake of completeness.

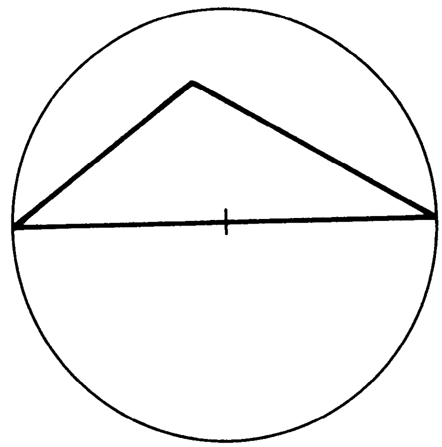


FIGURE II-1a. OBTUSE TRIANGLE - 2-POINT CIRCLE

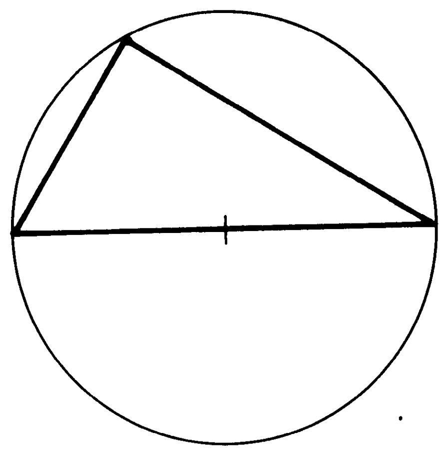


FIGURE 1b. RIGHT TRIANGLE - 2-POINT CIRCLE

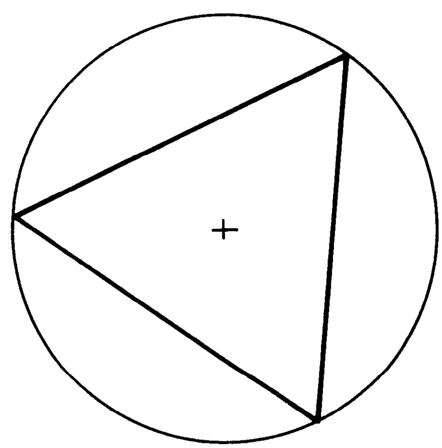


FIGURE 1c. ACUTE TRIANGLE - 3-POINT CIRCLE

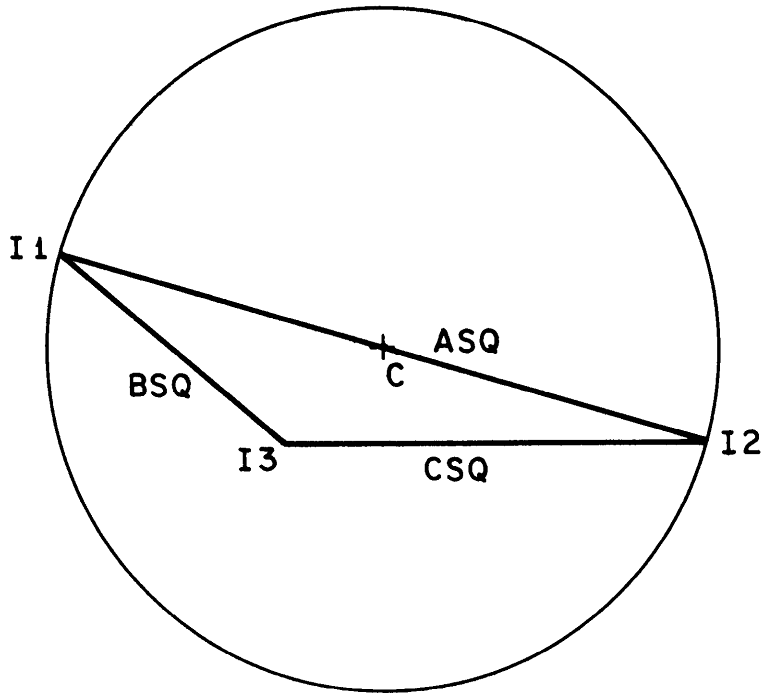


FIGURE II-2. NOMENCLATURE USED TO FIND THE CIRCUMSCRIBING CIRCLE

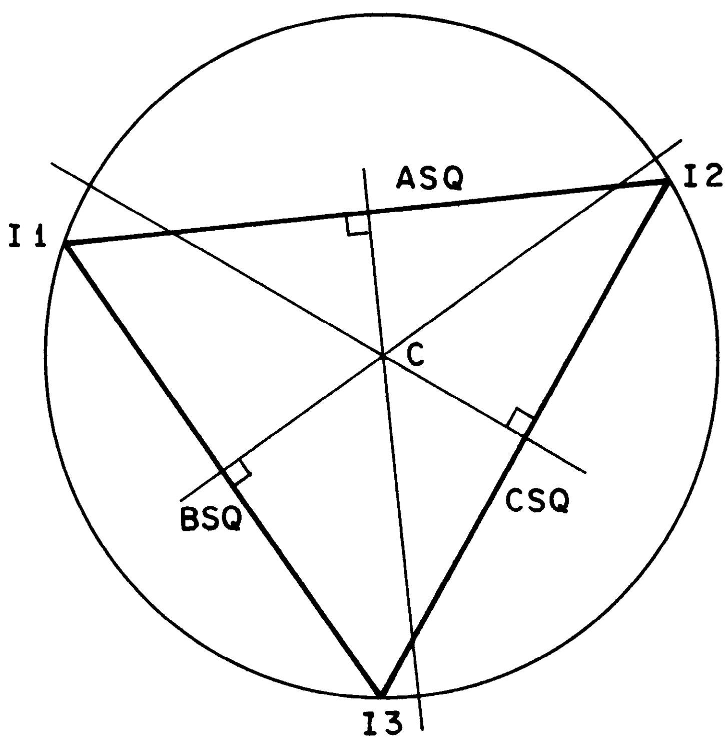


FIGURE II-3. CENTER OF A 3-POINT CIRCLE

If ASQ $\geq$ BSQ + CSQ, triangle is obtuse (II-4)

If ASQ < BSQ + CSQ, triangle is acute . (II-5)

As noted previously, if the triangle is obtuse, the CC is defined by the two points which determine the longest line. This line is the diameter of the CC. Knowing the end points of the diameter, the center and radius may be easily found.

If the triangle is acute, the relationship of the sides is given by Equation II-5. The center of the three point circle for this case may be found by solving a series of simultaneous equations, Figure II-3.

Given a set, N, of X, Y points defining an extruded cross section, any three points could be used for the first trial circle. In determining the CC of an extruded shape, the first trial points were set equal to 1, N/3 and 2N/3. This is an arbitrary choice. From these three points, the center and radius of the trial circle are found and then all remaining points are tested relative to this center and radius. When a point is found which lies outside the trial circle is found, a condition as shown in Figure II-4 exists. Point I4 is the point outside the trial circle. Using the new point, three new triangles can be constructed: (I1, I2, I4), (I1, I3, I4), and (I2, I3, I4).

Using the techniques described above, the nature (acute or obtuse), and then the center and radius of each CC for each of the three new triangles is found. The CC with the largest radius will enclose all four points. In Figure II-4, triangle (I1, I3, I4) is the new CC triangle with Center C' and radius R'. The entire process of testing all points relative to the new trial radius is then repeated.

Circle Circumscribing 3-Points

Given three points, the circle which passes through all three points can be found as follows:

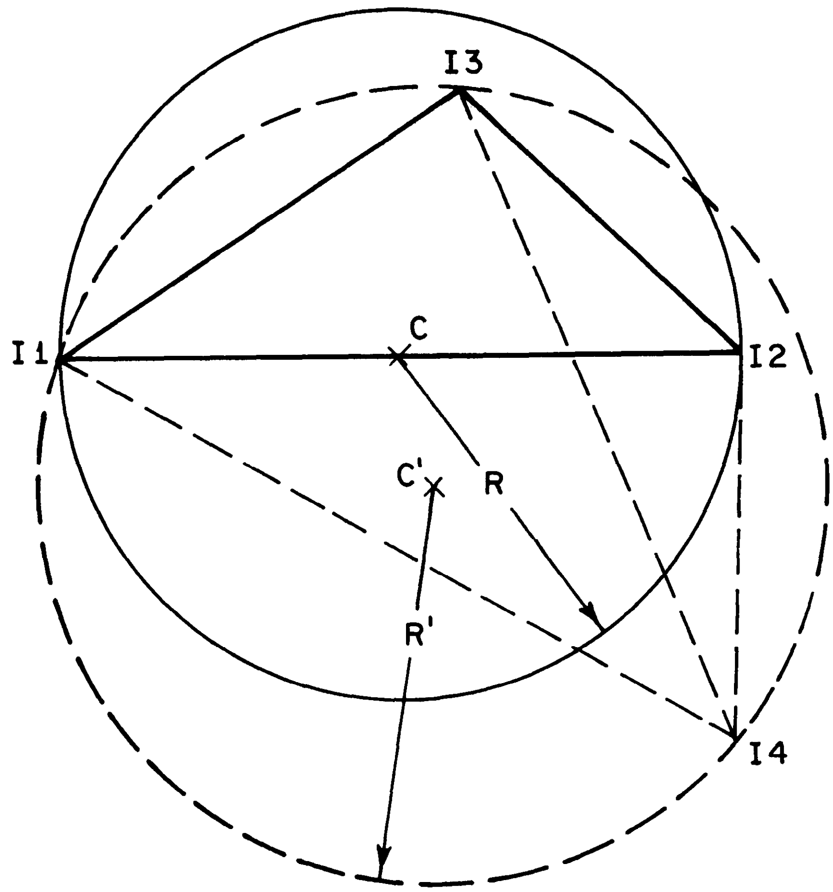


FIGURE II-4. GEOMETRIC CONDITION EXISTING WHEN A POINT LIES OUTSIDE THE CIRCLE WHICH CIRCUMSCRIBES THREE OTHER POINTS

The general equation of a circle is given by:

$$X^2 + Y^2 + C_1X + C_2X + C_3 = \emptyset \qquad \text{(II-6a)}$$

or

$$C_1X + C_2Y + C_3 = -(X_2 + Y^2) \qquad \text{(II-6b)}$$

Substituting three unique points for X and Y in Equation II-6b yields:

$$C_1X_1 + C_2Y_1 + C_3 = -(X_1^2 + Y_1^2) \qquad \text{(II-7a)}$$

$$C_1X_2 + C_2Y_2 + C_3 = -(X_2^2 + Y_2^2) \qquad \text{(II-7b)}$$

$$C_1X_3 + C_2Y_3 + C_3 = -(X_3^2 + Y_3^2) \qquad \text{(II-7c)}$$

Thus, II-7a, II-7b and II-7c are three equations in three unknowns. Using matrix algebra, the determinant of the system of equations can be found as:

$$\text{Det} = \begin{vmatrix} X_1 & Y_1 & 1 \\ X_2 & Y_2 & 1 \\ X_3 & Y_3 & 1 \end{vmatrix} \qquad \text{(II-8a)}$$

$$= X_1Y_2 + X_2Y_3 + X_3Y_1 - X_3Y_2 - X_2Y_1 - X_1\text{-}Y_3 \qquad \text{(II-9b)}$$

If Det $\neq 0$, a solution exists and the individual coefficients can be determined as:

$$C_1 = \frac{\begin{vmatrix} -(X_1^2 + Y_1^2) & Y_1 & 1 \\ -(X_2^2 + Y_2^2) & Y_2 & 1 \\ -(X_3^2 + Y_3^2) & Y_3 & 1 \end{vmatrix}}{\text{Det}} \qquad \text{(II-9a)}$$

or $$C_1 = [X_3^2 + Y_3^2)\ Y_2 + (X_2^2 + Y_2^2)\ Y_1 + (X_1^2 + Y_1^2)\ Y_3$$

$$- (X_1^2 + Y_1^2)\ Y_2 - (X_2^2 + Y_2^2)\ Y_3 + (X_3^2 + Y_3^2)\ Y_1]/\mathrm{Det} \qquad \text{(II-9b)}$$

C_2 and C_3 can be determined in a similar manner. Knowing the three coefficients, Equation II-6a can be rearranged into the form:

$$(X + \frac{C_1}{2})^2 + (Y + \frac{C_2}{2})^2 = - C_3 + C_1^2/4 + C_2^2/4 \qquad \text{(II-10)}$$

This is the common equation of a circle which is generally written as:

$$(X-A)^2 + (Y-B)^2 = R^2 \qquad \text{(II-11)}$$

From Equation II-10, the center of the circle is at $X = - C_1/2$, $Y = - C_2/2$, and has a radius of:

$$R = [.25(C_1^2 + C_2^2) - C_3]^{1/2} \quad . \qquad \text{(II-12)}$$

Calculation of the Area of a Cross Section

The area of any polygon, as shown in Figure II-5, may be obtained by the formula:

$$A_S = \frac{1}{2}\ (x_2y_1 - x_1y_2) + (x_3y_2 - x_2y_3) + \ldots + (x_ny_{n-1} - x_{n-1}y_n) + (x_1y_n - x_ny_1)$$

$$\text{(II-13)}$$

where $x_1, x_2, \ldots, x_n$ and $y_1, y_2, \ldots, y_n$ are coordinates of consecutive corners of the polygon with respect to a Cartesian coordinate system. A convenient choice of coordinate system in a nonsymmetrical part, as is the case with most aircraft structural parts, may originate at the parting line or at the datum line and one side of the piece.

The area bounded by two straight-lines and an arc of a circle, area abc, as shown in Figure II-6, may be calculated by:

$$A_R = R^2 \left(\tan \frac{\gamma}{2} - \frac{\gamma}{2}\right) \quad , \tag{II-14}$$

where R is the radius of the arc and γ is the included angle between two radii as defined in Figure II-6. If A_R is due to a fillet, then it is to be added onto the area of the polygon, A_S; on the other hand, if A_R is due to a convex corner, it should be subtracted from A_S.

Center of Gravity of a Cross Section

Center of gravity of the cross sections are used in the computation of the shape-complexity factor. The center of gravity of any polygon, as shown in Figure II-5, with respect to the y-axis is determined from the equation:

$$CG_S = \frac{\frac{1}{4}\left[\left(x_2^2 y_1 - x_1^2 y_2\right) + \left(x_3^2 y_2 - x_2^2 y_2\right) + \ldots + \left(x_n^2 y_{n-1} - x_{n-1}^2 y_n\right) + \left(x_1^2 y_n - x_n^2 y_1\right)\right]}{A_S} \tag{II-15}$$

where $x_1, x_2, \ldots, x_n$ and $Y_1, y_2, \ldots, y_n$ are the coordinates of the corners of the polygon, and A_S is calculated by Equation (II-13).

The center of gravity of an area, aebd, as shown in Figure II-7, bounded by two straight-lines and the arc of a circle may be calculated by:

$$\Delta x_i = R_g \cos \theta \quad , \tag{II-16}$$

where R_g is the radius to the center of gravity from the corner and θ is the angle between the direction of R_g and the x-axis, as shown in Figure II-7. Thus, when Δx_i is added into x_i, the location of the center of gravity of the area, aebd, is defined. Since

$$x_e = R\left(1 - \cos \frac{\gamma}{2}\right), \text{ and } x_c = R\left(\frac{1}{\cos \frac{\gamma}{2}}\right)$$

as defined in Figure II-7, then

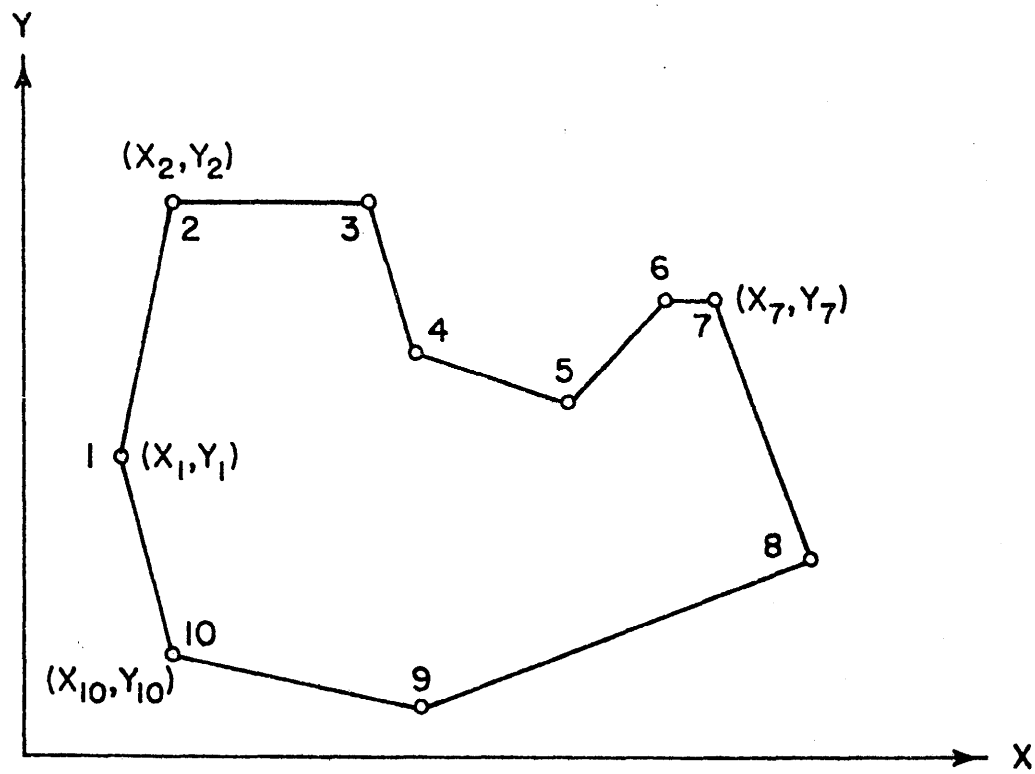


FIGURE II-5. DIAGRAM OF A POLYGON AND A RECTANGULAR COORDINATE SYSTEM DEFINING ITS CORNERS

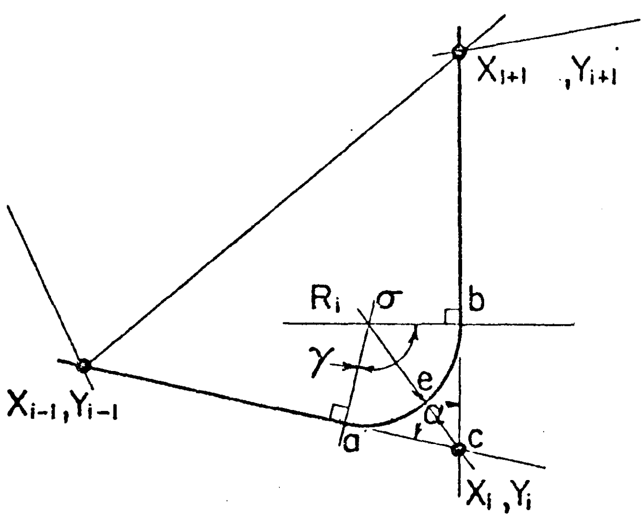


FIGURE II-6. SCHEMATIC DIAGRAM OF CORNER (i), ILLUSTRATING THE ADDITIONAL CROSS-SECTIONAL AREA DUE TO THE FILLET OF RADIUS R_i

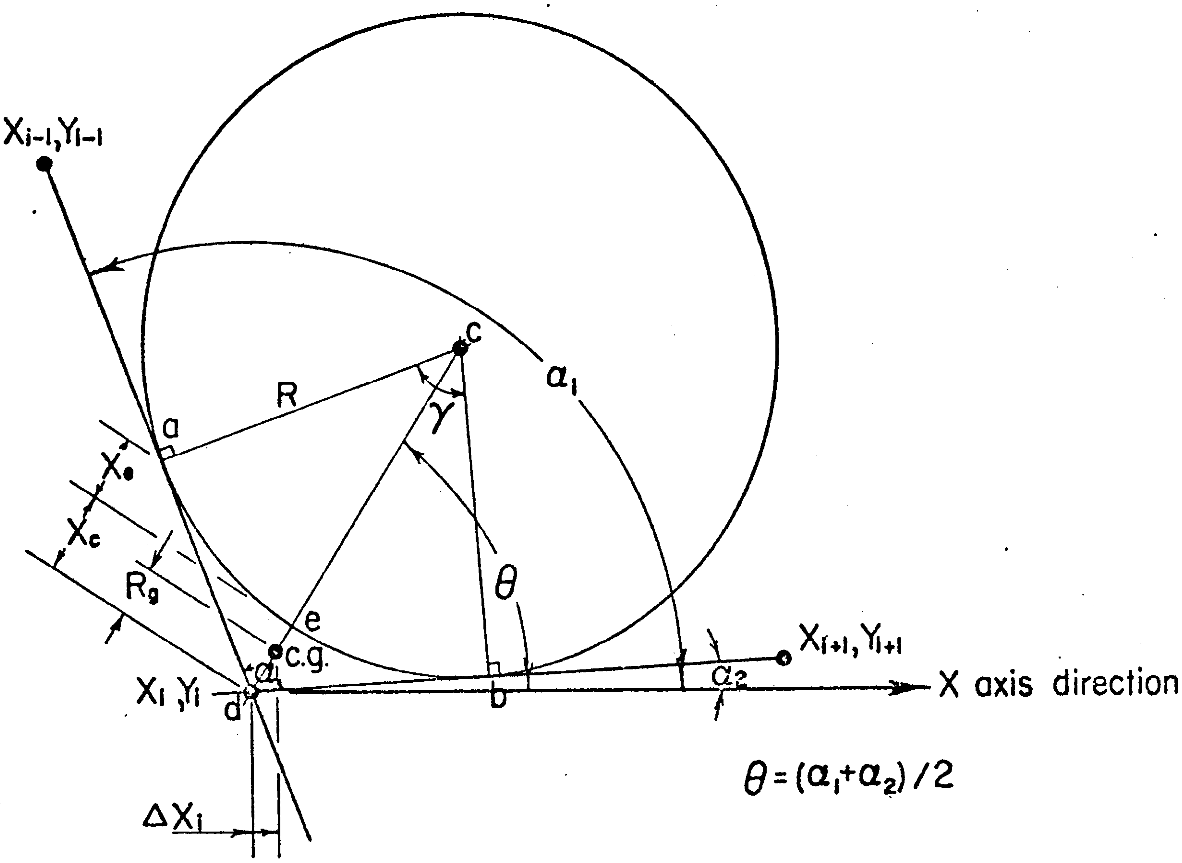


FIGURE II-7. SCHEMATIC DIAGRAM SHOWING THE LOCATION OF THE CENTER OF GRAVITY OF A FILLET AREA

$$R_g = \frac{\frac{2}{3} x_c \cdot A_T + (x_c + x_R)\ A_{RR}}{A_R} \qquad \text{(II-17)}$$

where

A_R is obtained from Equation (II-14)

$$A_T = x_c^{\ 2} \tan\left(\frac{\pi-\gamma}{2}\right) \qquad \text{(II-18a)}$$

$$A_{RR} = A_R - A_T \qquad \text{(II-18b)}$$

and

$$x_R = \frac{\frac{x_e^3}{3} \tan\frac{\alpha}{2} + \frac{x_c x_e^2}{2} \tan\frac{\alpha}{2} + \frac{(2Rx_e - x_e^2)^{3/2}}{3} - \frac{R}{2}\left[(x_e - R)\sqrt{2Rx_e - x_e^2} + \frac{R^2\gamma}{2}\right]}{A_{RR}/2} \ . \qquad \text{(II-19)}$$

The area, A_{RR}, given by Equation (II-18b) may also be expressed as:

$$A_{RR} = x_e^2 \tan\frac{\alpha}{2} + 2x_e x_c - (x_e - R)\sqrt{2Rx_e - x_e^2} - R^2\,\frac{\gamma}{2} \quad . \qquad \text{(II-20)}$$

The angle, α, as defined in Figure II-7, is:

$$\alpha = \pi - \gamma \ .$$

Perimeter of a Cross Section

The perimeter of any polygon, the coordinates of whose corners are $x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_n$, may be calculated by:

$$P_S = \left[\sum_{i=2}^{n} \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2}\right] + \sqrt{(x_n - x_1)^2 + (y_n - y_1)^2} \ . \qquad \text{(II-21)}$$

At each corner, this value must be reduced by:

$$R\gamma - 2R \tan\frac{\gamma}{2} \quad . \qquad \text{(II-22)}$$

This is due to the radius at that particular corner; therefore,

$$P = P_S + \sum_{i=1}^{n} |R_i| \, (\gamma - 2 \tan \frac{\gamma}{2}) \quad . \qquad \text{(II-23)}$$

Equation (II-23) will give the correct perimeter of a cross section, i.e., of a planar shape bounded by a succession of straight-lines and arcs of circles.

Fitting Circular Arcs to a Polygon

When working with planes defined as polygons with radii associated with each corner or fillet, there are a number of places where it is necessary to find the center of a radius, the tangency points of the arc with the polygon, or the subtended angle of the arc. Such instances occur in FITARC or DECUSP, for example. The following summarizes the mathematical derivations used in the programming of CENTER which finds the center and tangency points. All the symbols used are defined in Figure II-8, and this figure will be referred to implicitly throughout the following discussion.

Knowing the coordinates of the three points which define a corner or fillet of a polygon, the angles of each side can be found as:

$$\theta_B = \tan^{-1} [(Y_{i-1} - Y_i)/(X_{i-1} - X_i)] \qquad \text{(II-24)}$$

$$\theta_E = \tan^{-1} [(Y_{i+1} - Y_i)/(X_{i+1} - X_i)] \qquad \text{(II-25)}$$

The angle of the bisector can then be found as the average of θ_B and θ_E or

$$\theta_A = (\theta_B + \theta_E)/2 \quad . \qquad \text{(II-26)}$$

Since the center of the arc lies along the bisector and the tangency points intersect the sides of the polygon at right angles, half of the angle subtended by the arc can be found as follows:

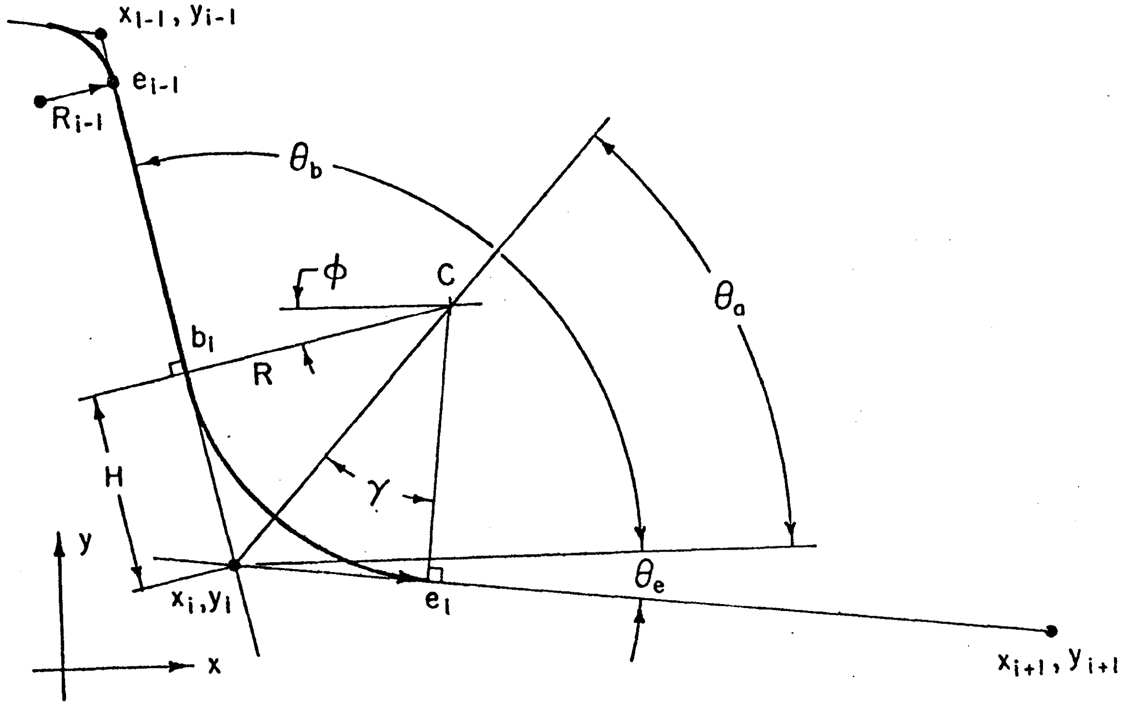


FIGURE II-8. FITTING AN ARC TO A POINT ON A POLYGON

θ_E is subtracted in the above expression since the value returned by the function is a signed quantity indicating both magnitude and direction.

Using expression for θ_A in the above gives:

$$\gamma = \theta_E - (\theta_B + \theta_E)/2 + \pi/2 \qquad \text{(II-28)}$$

$$\gamma = (\theta_E - \theta_B + \pi)/2 \quad . \qquad \text{(II-29)}$$

Since the total included angle approaches π as $(\theta_B - \theta_E)$ approaches π, the following adjustment is necessary:

$$\text{If } \gamma > \pi/2,\ \gamma = (\theta_E - \theta_B - \pi)/2 \quad . \qquad \text{(II-30)}$$

The absolute value of γ is the magnitude of the half-subtended angle and the sign of γ indicates the direction of rotation of the arc. If $\gamma > \phi$, the arc is counterclockwise. If $\gamma < \phi$, the arc is clockwise.

The center of the radius is next found as:

$$X_c = X_i + R\cos(\theta_A)/\cos\gamma \qquad \text{(II-31)}$$
$$Y_c = Y_i + R\sin(\theta_A)/\cos\gamma \quad . \qquad \text{(II-32)}$$

The distance, H, from the center to the tangency points can be expressed as:

$$H/R = \tan\gamma \text{ or } H = R\tan\gamma \quad , \qquad \text{(II-33)}$$

using the absolute value of γ.

The tangency points themselves are then found as:

$$X_B = X_C + H\cos\theta_B \qquad \text{(II-34)}$$

$$Y_B = Y_C + H\sin\theta_B \qquad \text{(II-35)}$$

and

$$X_E = X_C + H\cos\theta_E \qquad \text{(II-36)}$$

$$Y_E = Y_C + H\sin\theta_E \qquad \text{(II-37)}$$

For the purpose of graphically displaying an arc, it is necessary to simulate the arc as a series of short, linear segments. The smaller each of these segments are, the better the representation will be. The technique for calculating the end points of the individual segments is as follows:

With r = resolution desired in thousandths of an inch, the number of segments needed is the arc length divided by the resolution, or

$$n = \text{integer part of } [1000\ R\gamma/r + .5] \quad . \tag{II-38}$$

The angular increments can then be found as the total subtended angle divided by the number of increments, or

$$\delta = \frac{\gamma}{n} \quad . \tag{II-39}$$

The starting angle of the arc is found by knowing the center and the first tangent point as

$$\phi = \tan^{-1} [(Y_b - Y_c)/(X_b - X_c)] \quad . \tag{II-40}$$

Incrementing ϕ_b by the amount of δ for n times gives the coordinate points on the arc as:

$$X_{aj} = X_c + R \cos (\phi + j\delta) \tag{II-41}$$
$$Y_{aj} = Y_c + R \sin (\phi + j\delta) \tag{II-42}$$

$$\text{where} \qquad j = 0,1,2, \ldots n \quad . \tag{II-43}$$

APPENDIX III

COMPUTER HARDWARE CONFIGURATION, OPERATING SYSTEM SOFTWARE, AND PROGRAM ORGANIZATION

APPENDIX III

COMPUTER HARDWARE CONFIGURATION, OPERATING SYSTEM SOFTWARE, AND PROGRAM ORGANIZATION

Computer Hardware System

The ALEXTR and EXTCAM die-design and manufacturing systems have been developed and implemented on a Digital Equipment Corp (DEC) PDP-11/40 computer system. The computer is shown in Figure III-1. This computer is composed of the following elements:

(1) 32K (16-bit word) core memory. The upper 4K of memory is used for peripheral device registers leaving 28K available for the operating system and user software.

(2) High-speed paper-tape reader and punch.

(3) Two RK-11 disk-pack drives (1.2 million words each).

(4) LA-30 DEC-writer keyboard terminal.

(5) VT-11 refresh graphics display and control, including light pen.

(6) LPS data acquisition system. The analog output of this unit is used to drive the X-Y recorder for hard-copy plots.

(7) Hewlett Packard Model 7004B flat-bed X-Y recorder.

A special interface was built to adapt the LPS output control signals to the X-Y recorder.

Operating System Software

The programs were developed and operate using the following DEC software:

(1) RT-11 single-job operating system, version VØ2B-Ø5E

(2) FORTRAN-IV compiler, VØ1B - Ø80

(3) Link editor, VØ4 - Ø2

(4) MACRO assembler

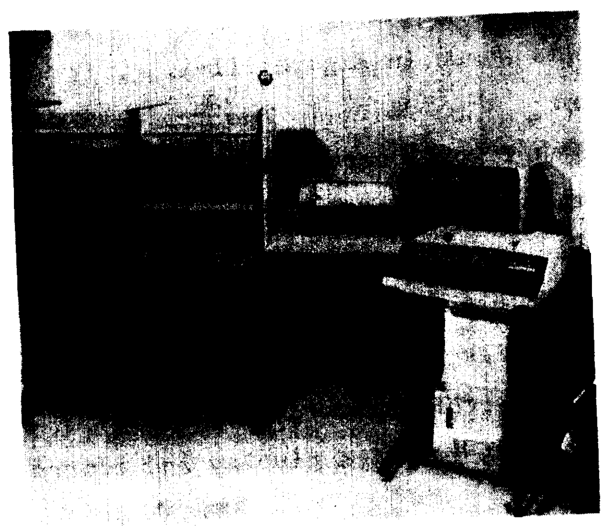


FIGURE III-1. DEC PDP-11/40 COMPUTER USED TO DEVELOP ALEXTR AND EXTCAM EXTRUSION DIE CAD/CAM SYSTEMS

Both ALEXTR and EXTCAM have been written in FORTRAN-IV with the exception of the graphics display and plotter handlers in ALEXTR, which are in MACRO assembler. Both systems are linked as one root segment with two overlay regions. The upper memory address used by ALEXTR (excluding the RT-11 operating system) is approximately 17451 (base 10 words). The two overlay regions require 2811 words; the combined length of all segments which run in these two regions is approximately 20,881 words. The general organization of memory is given in Figure III-2. The organization of the program modules of ALEXTR is given in Table III-1 and Figure III-3, and for EXTCAM in Table III-2 and Figure III-4.

Compiling and linking is performed following the directions in DEC manual No. DEC-11-LRFPA-A-D, "RT-11 FORTRAN Compiler and Object Time System User's Manual". When compiling, the root segment should be the last module compiled, and it should be the only module which uses the "U" compiler switch.

Linking the object modules to create the save-image file with the overlay structure is normally done using the BATCH processor of RT-11. The link commands for BATCH are contained in file LNKEXT.BAT and EXTCAM.BAT for ALEXTR and EXTCAM, respectively, and are listed in Figures III-5 through III-8. The files FOREXT.BAT and FORCAM.BAT are available for BATCH compiling all program segments.

Before running either program, the monitor directive "GT ON" is usually specified to indicate that the GT-40 display is to be used as the output terminal.

The overlay structure created by the linker is such that all Region 2 overlays are positioned above the top of the largest segment in the Region 1 overlay area. Thus, the total overlay area required is the sum of the largest segments in each region. From Table III-2, it can be seen that total core required for overlays in ALEXTR is governed by segments OVRL1G in Region 1, and OVRL2G in Region 2. It has been found that an upper limit of 21,000 (base 10 words; 122,000 base 8 bytes) is approximately the maximum allowable program core-load size. Care should be taken in modifying any segments to ensure that the segments do not become larger than this upper limit.

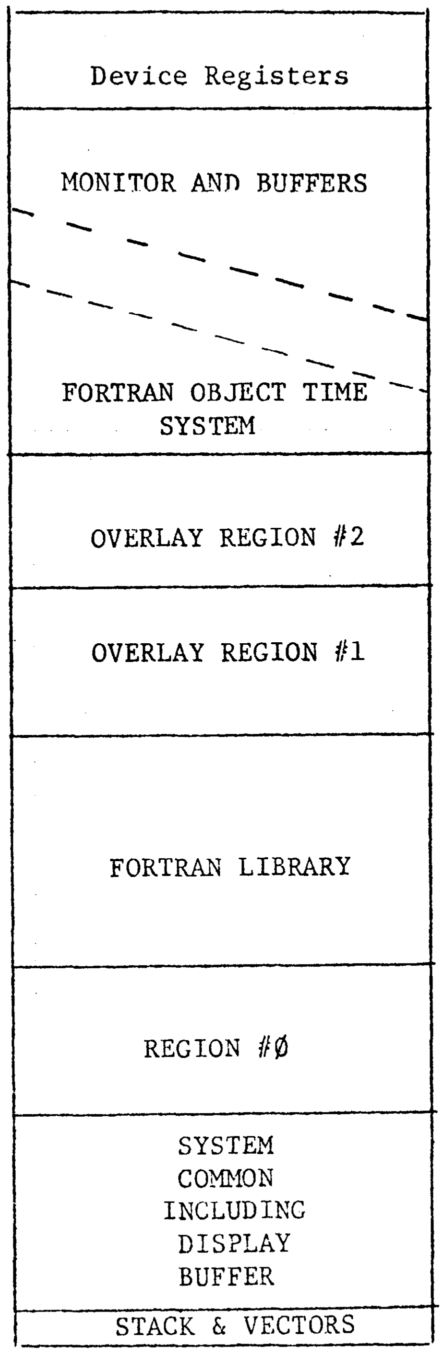


FIGURE III-2. RUN TIME MEMORY ORGANIZATION TYPICAL FOR ALEXTR AND EXTCAM

TABLE III-1. ORGANIZATION OF ALEXTR PROGRAMS

Overlay Region No.	Module No.	Module Name	Module Size(1)	Element Name	Element Type(2)	Code(3)
Ø	1	ALEXTR	6375	ALEXTR	P	F
				SYSPR	C	F
				SECTN	C	F
				DISPLA	C	F
				NTRPID	C	F
				PLACES	C	F
				BERNGS	C	F
				SPARE1	C	F
				SCALZZ	S	F
				ACOS	F	F
				TAN	F	F
				CENTER	S	F
				MARKXX	S	F
				MARKIT	S	F
				DISTSQ	F	F
Ø	2	GRPHCS	842	GT40	G	M
				VTHDLR(*)	G	M
1	1	OVRL1A	952	PRPROS	S	F
				READIN	S	F
				PARAMS	S	F
				XSECA	S	F
1	2	OVRL1B	915	PICTUR	S	F
				INITDS	S	F
1	3	OVRL1C	1088	ERRPRT	S	F
				SETUPS	S	F

TABLE III-1. (Continued)

Overlay Region No.	Module No.	Module Name	Module Size(1)	Element Name	Element Type(2)	Code(3)
1	4	OVRL1D	1007	ENDIT	S	F
				PRSCHR	S	F
1	5	OVRL1E	1319	DIHOLE	S	F
1	6	OVRL1F	1506	PLACEM	S	F
				STORIT	S	F
1	7	OVRL1G	1521	MODLYO	S	F
1	8	OVRL1H	1266	DICAVE	S	F
1	9	OVRL1J	1252	DISTRS	S	F
1	10	OVRLkL	759	BEARNG	S	F
2	1	OVRL2A	734	INTRPL	S	F
				FITARC	S	F
				PLOTIT	S	F
				MAXMIN	S	F
	1	PLOTOB	658	PLOTOB	S	M
2	2	OVRL2B	1283	CRCMCL	S	F
				CRCLPR	S	F
				FNDCRC	S	F
				MATERL	S	F
2	3	OVRL2C	1049	LOAD	S	F
				RUNIT	S	F
				REPRT1	S	F
				REPRT2	S	F
2	4	OVRL2S	1032	XYINTR	S	F
				DRWSEG	S	F
				MARKCG	S	F
				ALLSEG	S	F

TABLE III-1. (Continued)

Overlay Region No.	Module No.	Module Name	Module Size(1)	Element Name	Element Type(2)	Code(3)
				PLTSEG	S	F
				DRWCRC	S	F
				UNSCAL	S	F
2	5	OVRL2E	1223	TESTIT	S	F
				ROTMIR	S	F
				SHFTIT	S	F
				GETHIT	S	F
2	6	OVRL2F	1169	TRNSLT	S	F
				TRNSIT	S	F
				TNGSTR	S	F
2	7	OVRL2G	1290	AREAX	S	F
				GETPTS	S	F
				TOOLCH	S	F
				LYONEW	S	F
				ENDFLS	S	F
2	8	OVRL2H	868	ADDPNT	S	F
				LINEQ	S	F
				TNGCMP	S	F

(1) Size in base 10 words.

(2) Module Type - C = Common, F - Function, G = Group of FORTRAN callable subroutines, P = Program, S = Subroutine.

(3) Code - F = FORTRAN, M = Macro Assembler.

(*) Object code supplied by DEC with VT-11 display system hardware.

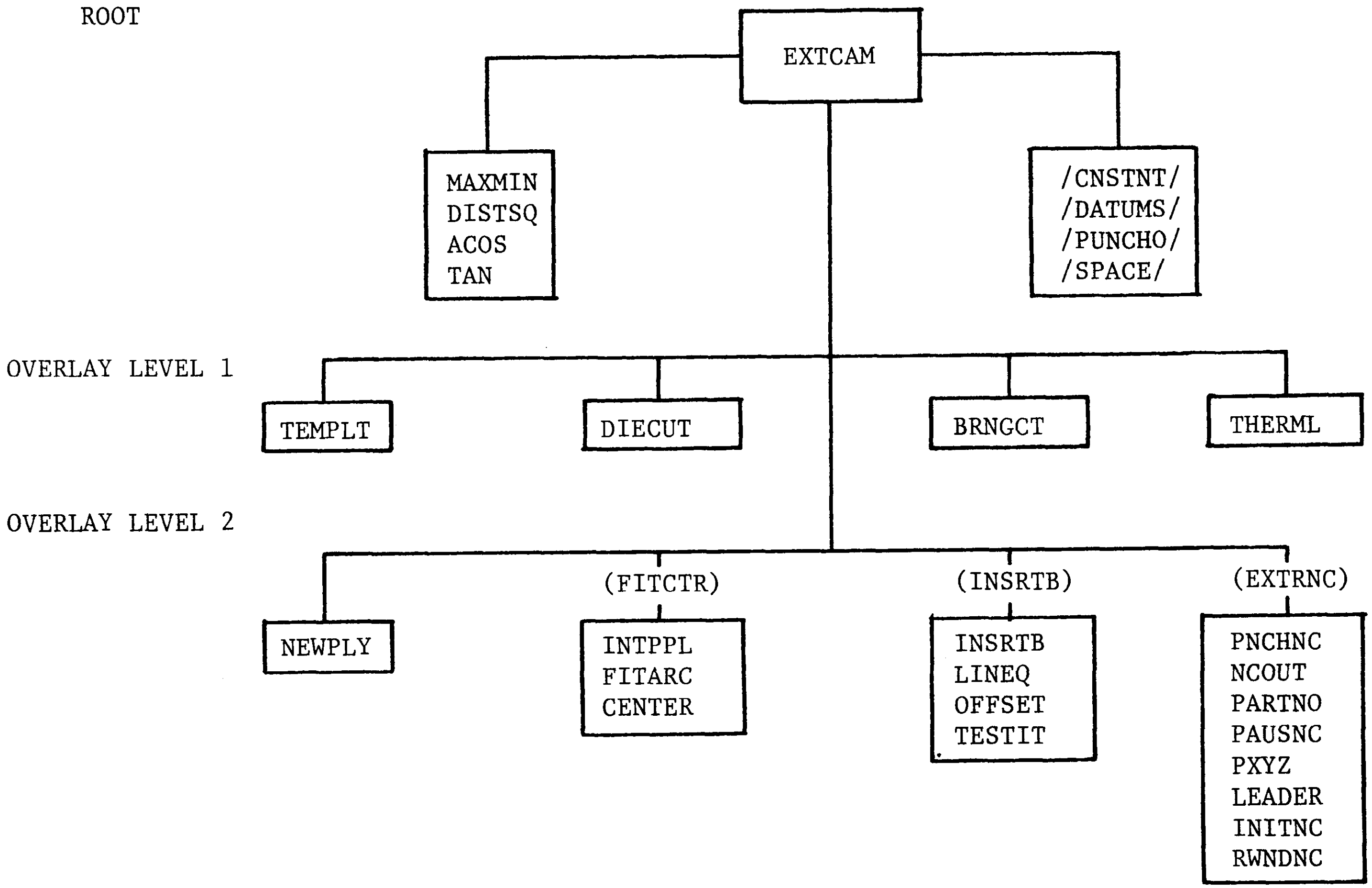


FIGURE III-4. EXTCAM PROGRAM STRUCTURE

TABLE III-2. ORGANIZATION OF EXTCAM PROGRAMS

Overlay Region No.	Module No.	Module Name	Module Size(1)	Element Name	Element Type(2)
Ø	1	EXTCAM	5017	EXTCAM	P
				MAXMIN	S
				DISTSQ	F
				ACOS	F
				TAN	F
				CNSTNT	C
				DATUMS	C
				PUNCHO	C
				SPACE	C
1	1	TEMPLT	969	TEMPLT	S
1	2	DIECUT	1055	DIECUT	S
1	3	BRNGCT	3126	BRNGCT	S
1	4	THERML	321	THERML	S
2	1	NEWPLY	407	NEWPLY	S
2	2	FITCTR	725	INTRPL	S
				FITARC	S
				CENTER	S
2	3	INSRTB	1027	INSRTB	S
				LINEQ	S
				OFFSET	S
				TESTIT	S
2	4	EXTRNC	862	INITNC	S
				LEADER	S
				NCOUT	S
				PARTNO	S
				PAUSNC	S
				PNCHNC	S
				PXYZ	S
				RWNDNC	S

(1) Size in words (base 10)

(2) Element Type - C = Common, F = Function, P = Program, S = Subroutine.

All program elements are coded in FORTRAN.

```
$JOB
$MESSAGE EXTCAM MODULES BATCH COMPILE
$RT11
.RUN FORTRA
*TEMPLT=TEMPLT/E/W
*DIECUT=DIECUT
*BRNGCT=BRNGCT
*THERML=THERML
*FITCTR=FITCTR
*NEWPLY=NEWPLY
*INSRTB=INSRTB
*EXTRNC=EXTRNC
*EXTCAM=EXTCAM/U
$EOJ !BATCH COMPILE DONE
```

FIGURE III-7. BATCH COMPILE FILE FOR EXTCAM

```
$JOB
$MESSAGE EXTCAM BATCH LINK
$RT11
.R LINK
*EXTCAM,EXTCAM=EXTCAM/F/B:11ØØ/C
*TEMPLT/O:1/C
*DIECUT/O:1/C
*BRNGCT/O:1/C
*THERML/O:1/C
*NEWPLY/O:2/C
*FITCTR/O:2/C
*INSRTB/O:2/C
*EXTRNC/O:2
$EOJ !BATCH LINK DONE
```

FIGURE III-8. BATCH LINK FILE FOR EXTCAM

```
$JOB
$MESSAGE EXTRUSION MODULES BATCH COMPILE
$RT11
.RUN FORTRA.SAV
*SYØ:OVRL1A=SYØ:OVRL1A/W
*SYØ:OVRL1B=SY0:OVRL1B
*SYØ:OVRL1C=SYØ:OVRL1C
*SYØ:OVRL1D=SYØ:OVRL1D
*SYØ:OVRL1E=SYØ:OVRL1E
*SYØ:OVRL1F=SYØ:OVRL1F
*SYØ:OVRL1G=SYØ:OVRL1G
*SYØ:OVRL1H=SYØ:OVRL1H
*SYØ:OVRL1J=SYØ:OVRL1J
*SYØ:OVRL1K=SYØ:OVRL1K
*SYØ:OVRL2A=SYØ:OVRL2A
*SYØ:OVRL2B=SYØ:OVRL2B
*SYØ:OVRL2C=SYØ:OVRL2C
*SYØ:OVRL2D=SYØ:OVRL2D
*SYØ:OVRL2E=SYØ:OVRL2E
*SYØ:OVRL2F=SYØ:OVRL2F
*SYØ:OVRL2G=SYØ:OVRL2G
*SYØ:OVRL2H=SYØ:OVRL2H
*SYØ:ALEXTR=SYØ:ALEXTR/U
$EOJ !BATCH COMPILE DONE
```

FIGURE III-5. BATCH COMPILE FILE FOR ALEXTR

```
$JOB
$MESSAGE EXTRUSION MODULES BATCH LINK
$RT11
.R LINK
*SYØ:ALEXTR,ALEXTR.MAP=SYØ:ALEXTR,GRPHCS/F/B:12ØØ/C
*SYØ:OVRL1A/0:1/C
*SYØ:OVRL1B/0:1/C
*SYØ:OVRL1C/0:1/C
*SYØ:OVRL1D/0:1/C
*SYØ:OVRL1E/0:1/C
*SYØ:OVRL1F/0:1/C
*SYØ:OVRL1F/0:1/C
*SYØ:OVRL1H/0:1/C
*SYØ:OVRL1J/0:1/C
*SYØ:OVRL1K/0:1/C
*SYØ:OVRL2A,PLOTOB/0:2/C
*SYØ:OVRL2B/0:2/C
*SYØ:OVRL2C/0:2/C
*SYØ:OVRL2D/0:2/C
*SYØ:OVRL2E/0:2/C
*SYØ:OVRL2F/0:2/C
*SYØ:OVRL2G/0:2/C
*SYØ:OVRL2H/0:2
```

FIGURE III-6. BATCH LINK FILE FOR ALEXTR

APPENDIX IV

ABBREVIATIONS

L.P: Light pen

PDP-11/40: The model of the DEC computer used for the ALEXTR and EXTCAM system.

APPENDIX IV

ABBREVIATIONS

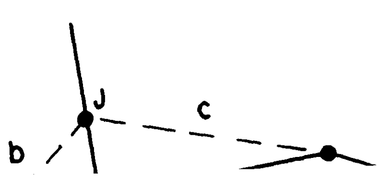
The following abbreviations are used in describing the operation and use of the subroutines in the ALEXTR and EXTCAM extrusion die design and manufacturing systems.

ALEXTR:	A system of computer programs for the design and analysis of flat-face aluminum extrusion dies.
CAD/CAM:	Computer-aided design and computer-aided manufacturing
CC:	Circumscribing circle -- the smallest circle which will completely enclose a given set of coordinate points
CCC:	Center of the circumscribing circle
CCD:	Circumscribing circle diameter. The diameter of the smallest circle which will completely enclose a given set of coordinate points
CD:	Center of the die
CG:	Center of gravity
CR:	Carriage return key on terminal keyboard
CRT:	Cathode ray tube display device
DEC:	Digital Equipment Corporation -- the manufacturer of the computer on which the ALEXTR and EXTCAM systems were developed
E.O.F:	End of file. A mark designating the end of a data file
E.O.S:	A card designating the end of the data defining a section polygon
LB:	Light button - a piece of text displayed on the CRT which is light pen sensitive

APPENDIX V

DESCRIPTION OF COMPUTER PROGRAMS FOR ALEXTR

V-2



APPENDIX V

DESCRIPTION OF COMPUTER PROGRAMS FOR ALEXTR

Function ((ACOS(X))

This function returns the arc cosine of X. That is, the value returned is the angle, in radians, whose cosine is X. This value is determined as follows:

1. If X = +1. , ACOS(X) = Ø.
2. If X = -1. , ACOS(x) = π
3. If -1. < X < +1. , ACOS(X) = $\mathrm{Tan}^{-1}\ ((1 - X^2)^{1/2}/X)$

This program element resides in the root segment as part of module ALEXTR.

Subroutine ADDPNT (JX,JY,K1,K2,IFLAG)

ADDPNT adds a point to an array of points which define the perimeter of an extrusion. The point is inserted where it lies closest to an existing line connecting two adjacent points in the array. If where the point is to be added does not lie between points K1 and K2, an error condition exists and the point is not added.

Where the point is to be added is found by first determining the length of the sides of the triangles formed by the point to be added and each pair of adjacent array points. Referring to Figure V-2, let 1,2,3, and 4 represent points in the existing array and J be the point to be added. The three sides, a, b, and c of triangle 2-3-J are found Then if the triangle is obtuse and

$$\mathrm{abs}\ (c^2 - b^2) \leq a^2 \quad , \tag{V-1}$$

J lies between 2 and 3. The normal distance, d, from J to line 2-3 nce, d, is then found. This is done by obtaining the coefficients to the coordinates of point J. The distance d is compared to the previous normal, and if it is a new minimum, the index to point 2 is saved.

(1) New Section: reads in data for section specified and calculates section parameters.

(2) New Press: gets parameters for press and alloy to be used.

(3) Load: Calculates load on die and determines yield and production rate.

(4) Locate: positions the extrusion openings on the die, and permits die design, stress analysis, and bearing size specification.

(5) Done: terminates the ALEXTR system.

Other than this query, all other executable statements in ALEXTR are calls to subroutines at lower overlay levels.

Another major function of this subroutine is to establish communication between various program elements by means of Common statements. Variables which are defined in the Common blocks of ALEXTR can be accessed by any other program module, regardless of module type or position in the overlay structure. Elements in Common which are constants, such as π, are also defined in ALEXTR by means of Data statements.

Subroutine ALLSEG(N)

ALLSEG determines the position of the center of gravity for all segments of a multi-hole die, based on the location of the CG of the first segment. Knowing the position of the first CG in cartesian coordinates relative to the center of the die, its position in polar coordinates is found as:

$$R = (X_1^{\ 2} + Y_1^{\ 2})^{.5} \tag{V-2}$$

$$\beta_1 = \tan^{-1} (Y_1/X_1) \tag{V-3}$$

The included angle of each segment, α, is determined by

$$\alpha = 2\ \pi/N \tag{V-4}$$

where N is the number of openings.

When all triangles have been tested, the index to the first point of the line closest to the point to be added is compared to indices K1 and K2. The index N must meet the following criteria:

$$K1 \leq N \leq K2 \quad .$$

If N is not within this range, a return is made with the flag set to 1. If N is within the range, all points from L + 1 are shifted down one position in the array. The length from the first end point to the new point is found by the difference of the squares of the other two sides of the triangle formed by the first end point, the new point as input, and the new point on the line. A ratio is then generated as the distance to the new point on the line A from the first end point, and the length of the line between the two end points. This ratio is found using CRT raster dimensions. The ratio is then applied to the corresponding points in user space to locate the new point in the users dimensions. The radius at the new point is arbitrarily set equal to 0.5 inch.

The parameters in the calling sequence are:

(1) JX,JY: Coordinates of the new point to be added (input)

(2) K1,K2: Indices in the array of points that JX,JY is to be added ti

(3) IFLAG: A flag. If = 1, error.

ADDPNT is part of module OVRL2H.

Program ALEXTR

This is the root program segment of the entire extrusion die CAD/CAM system. It does no arithmetic processing or calculations. Rather, its sole purpose is to serve as a conduit to the various overlay segments. It essentially consists of nothing but a query to the user as to what operation is to take place:

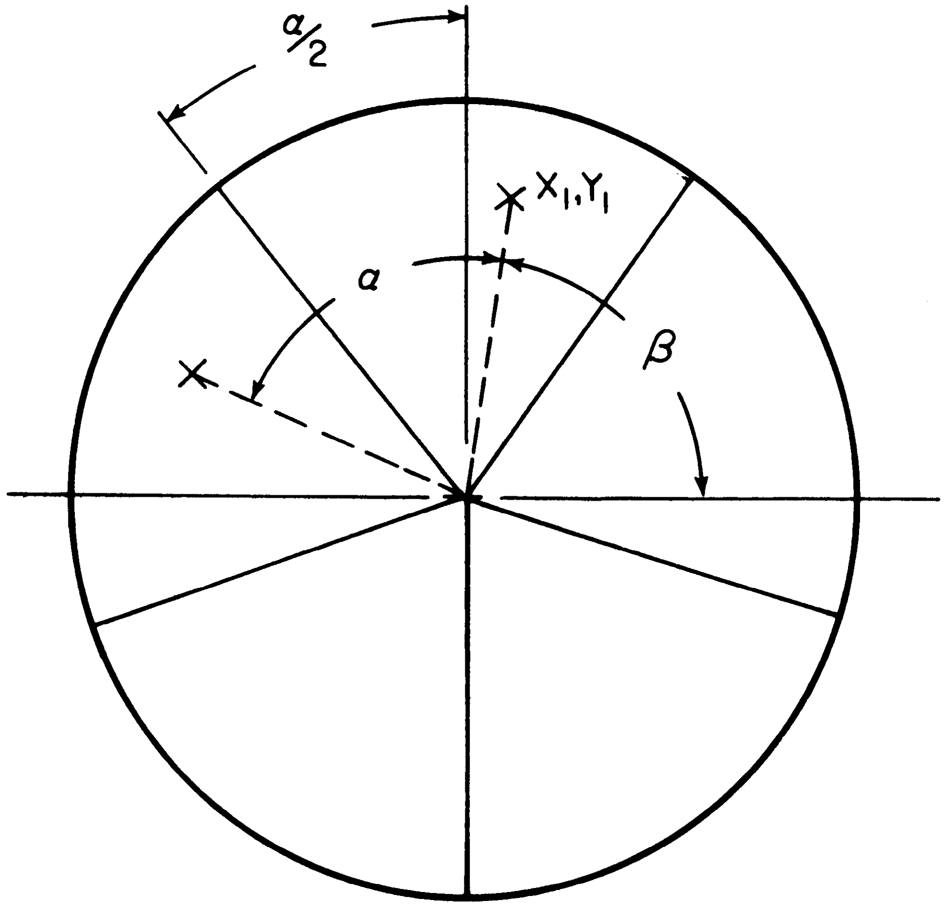


FIGURE V-2. GEOMETRY USED TO LOCATE MULTIPLE OPENINGS

To obtain the position of the second CG, β is incremented by α or

$$\beta_2 = \beta_1 + \alpha \qquad \text{(V-5)}$$

$$X_2 = (R) \cos (\beta_2) \qquad \text{(V-6)}$$

$$Y_2 = (R) (\sin \beta_2) \qquad \text{(V-7)}$$

These geometric relationships are shown in Figure V-2.

The mirror image parameter for the segment is also equated to the mirror image parameter for the first opening.

After the CG parameters are determined, the single segment display is removed. The full die circle is restored to view and then the segment separators are drawn. To simplify the programming, these are drawn as incremental vectors relative to the die center. They are drawn visible from the center out to the die parameter and invisible when returning to the center.

ALLSEG is part of module OVRL2D.

Subroutine AREAX (X,Y,N,AREA)

AREAX determines the area bounded by the set of N coordinates X,Y. It is similar to XSECA in operation. However, AREAX determines only the area, while XSECA determines the area, perimeter and center of gravity.

AREAX is part of module OVRL2G.

Subroutine BEARNG

BEARNG developes data regarding the bearing length of the die. The bearing is the thickness of the die across which the metal flows as it extrudes. The length of the bearing has a significant effect on the resistance seen by the material as it is forced through the die. The longer the bearing, the greater the resistance. This effect of bearing length on flow resistance is

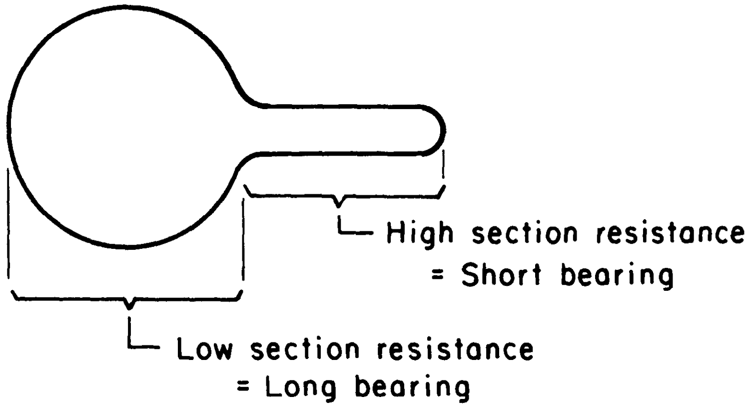


FIGURE V-3. RELATIONSHIP OF DIE BEARINGS AND SECTION AREAS

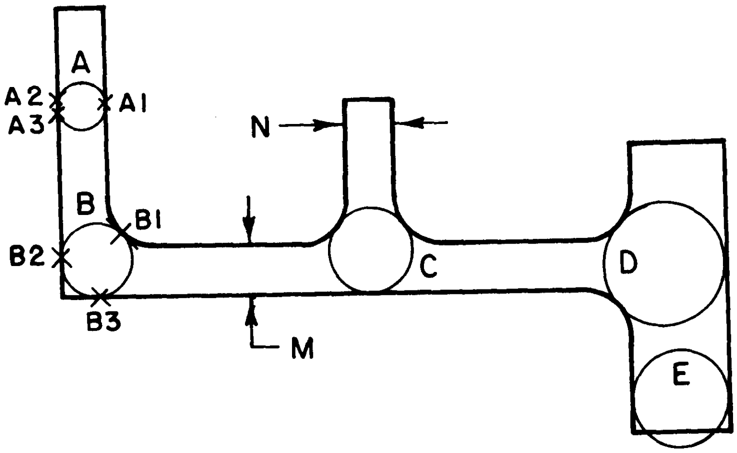


FIGURE V-4. IDENTIFYING SECTION THICKNESSES FOR DIE BEARING DETERMINATION

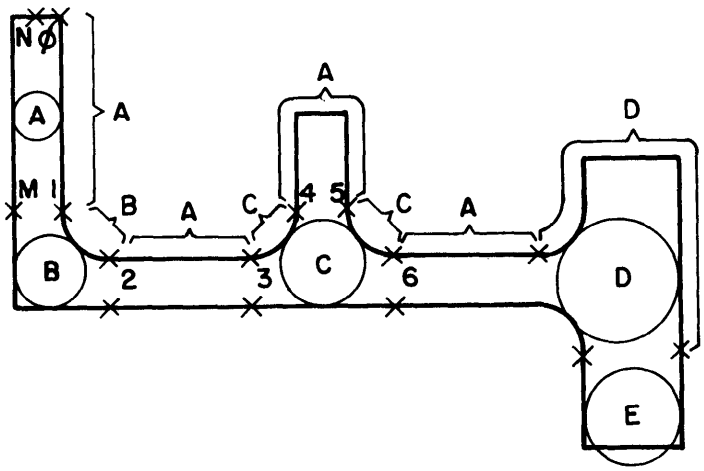


FIGURE V-5. RELATING SECTION THICKNESSES TO DIE PERIMETER POSITIONS capitalized on by varying the bearing length according to the thickness of the section at a particular location. For example, the round area of the extrusion shown in Figure V-3 would have less resistance than the rectangular fin. In order to have uniform resistance and, therefore, uniform flow across the entire die, the bearing in the round section would be made longer than the bearing in the fin. If the flow resistance is not uniform, material in the round area would flow faster than material in the fin. This would result in the material hooking as it extrudes, and/or a lack of complete fill in the fin.

BEARNG allows the user to indicate the relative thicknesses of an extrusion and the portions of the perimeter over which these thicknesses are to act. This is done in two steps. Identifying the thickness of various sections is done by first making the entire perimeter of the opening light pen sensitive. The user then uses the light pen to indicate three points on the perimeter. These three points are used to generate a three-point circle. The diameter of this circle is used as the thickness of the opening at that particular position. The user must ACCEPT each point as he locates it. After the third point is accepted, the X's marking the points picked are erased and the thickness circle is drawn. After a thickness circle is drawn, this phase of BEARNG may be terminated by picking END instead of locating another perimeter point. The circle size is found using CRCMCL and is drawn using DRWCRD.

This procedure is shown in Figure V-4. Points A1, A2, and A3 would be located using the light pen and ACCEPTing each point. After A3 was ACCEPTed, circle A would be determined and drawn. Similarly, points B1, B2, and B3 would be located by the user and then used to generate circle B. It should be noted that on the actual CRT display, the points used to generate each circle are erased when the circle is drawn. As many circles as needed to describe the different section thicknesses may be generated. In Figure V-4, the five circles A, B, C, D, and E would be sufficient. No circle was generated at M or N, as the thickness at these points is the same as where circle A was generated and circle A may be used to define the thickness at M and N.

The position of the three points used for each circle is maintained in absolute CRT raster units. CRCMCL is used to find the center and diameter

Subroutine CENTER
(X,Z,R,XC,ZC,ALPHA,SGN)

Given three points defining an angle and the radius which is to be fitted to the angle, CENTER determines the location of the center and the two points of tangency. The details of the procedures used are given in Appendix II. The parameters passed in the calling sequence are as follows:

(1) X,Z: One-dimensional arrays, of three elements each, defining the coordinate points of the angle (input).

(2) R: An array of three elements where the second element defines the radius to be fitted to the angle (input).

(3) XC,ZC: Three-dimensional arrays where the first elements are the coordinates of the center of the radius, and the second and third elements are the coordinates of the tangency points (output).

(4) ALPHA: The angle subtended by the arc to be fitted (output).

(5) SGN: Sign of the sense of rotation of the arc; positive if counter-clockwise, negative if clockwise (output).

CENTER is part of the root segment module ALEXTR.

Subroutine CRCLPR

(XI,YI,X2,Y2,X3,Y3,XC,YC,R)

Given three points, CRCLPR finds the center and radius squared of the circle which passes through all three points. A complete description of the mathematics used is given in Appendix II. The variables in the calling sequence are

(1) X1,Y1,X2,Y2,X3,Y3: Coordinates describing three points (input).

of the circle, also in raster units. DRWCRC is then called to draw the circle, after temporarily shifting the origin of the display to the center of the calculated circle. The circle diameter is converted to the user's units after the circle is drawn.

Once all required thickness circles have been located, it is then necessary to identify over what portion of the perimeter opening each thickness is applicable. This is done by locating a point on the surface with the light pen. After the point is ACCEPT'ed, a circle must be picked and ACCEPT'ed.

These data are later interpreted in EXTCAM to mean that the latest circle size (section thickness) picked is to be used to determine the bearing length between the latest perimeter point picked and the previous perimeter point.

Referring to Figure V-5, point 1 would be located and then circle A would be picked. This would indicate that along the perimeter between points Ø and 1, the bearing is to be based on a thickness which is equal to the diameter of circle A. After point 2 was located, circle B would be picked as the bearing thickness between points 1 and 2. When point 3 is located, circle A is again chosen as the thickness for the bearing length. The last point, N, on the perimeter may be located anywhere. The last circle picked is used to determine the bearing from the previous point, M, back to the origin, regardless of where N is located. After the last thickness circle is identified, END is picked to terminate BEARNG.

Messages are printed at each step to prompt the user as to whether he is to locate a point on the perimeter or identify a circle. ACCEPT must be picked after either operation in order to have the operation completed.

The points on the perimeter are returned as absolute CRT raster units. When all points and circles have been selected, the perimeter points are converted to the user's units in the user's coordinate system using UNSCAL.

No parameters are passed to BEARNG in the calling sequence. The data for the location and size of each bearing is returned through the named common /BERNGS/.

BEARNG is contained in module OVERL1K.

CRCMCL is part of module OVRL2B.

Subroutine DICAVE

DICAVE provides a means for compensating long, narrow single-hole dies against the tendency for the die opening to close under load. Because the center sections of such a die are (relatively) considerably further from the edge than the ends of the opening, the die itself will tend to elastically deform near the center. This will tend to close the opening and produce an extrusion section which is thinner than specified.

When first called, DICAVE copies the extrusion polygon points (the original data input) to a second array which will represent the die opening polygon. All subsequent modifications to the data, whether in this subroutine or others, is performed on the die opening polygon data arrays. If a single opening has been specified and the user wishes to provide cave compensation, he is asked to enter the cave allowance. A default value is also provided.

The points defining the polygon are then displayed as light pen sensitive points, with the first two points also indicated with X's. The user is then asked to indicate the two points which will define the cave axis. This is done using the light pen.

Subroutine GETHIT is used to identify the polygon points picked. A line is drawn on the CRT between the two points picked to indicate the axis and a second line is drawn normal to this axis through the center of display. An example of the CRT display at this point is shown in Figure V-6. In this figure, A and B represent the points picked by the user to define the ends of the cave axis. The short dashed line is the cave axis; the the long, short dashed line is the normal to the axis.

The user is then asked if he wishes to add a new point to the die polygon. If the two points defining the cave axis are consecutive, he should answer Y(yes), or the cave compensation will have no effect. To add a new point, the entire perimeter of the section is made light pen sensitive. When the user touches the light pen to the perimeter, an "X" is drawn. This X will follow the light pen as it is moved along the perimeter. The coordinates of the X are determined when ACCEPT is touched. GETHIT is used to draw the X

(2) XC,YC: The coordinates of the center of the circle which circumscribes the three points (output).

(3) R: The radius squared of the circumscribing circle.

CRCLPR is part of module OVRL2B.

Subroutine CRCMCL

(X,Y,NI,CCD,XC,YC,I1,I2,I3)

Given a set of coordinate arrays X, Y of length NI, this subroutine will determine the smallest circle which will enclose (circumscribe) the points. A complete description of the mathematics involved to make this determination is given in Appendix II. In summary, the routine operates on the following logic.

(1) Pick three points for first test.

(2) Determine center and radius of test circle.

(3) Determine if any points lie outside test circle.

(4) If a point is outside the test circle
   (a) Find size of three new circles.
   (b) Set largest as new test circle.
   (c) Go back to (3), above.

(5) If no point is outside the test circle, test circle is circumscribing circle. Return diameter and center.

The parameters in the calling sequence are as follows:

(1) X,Y: Arrays of coordinate pairs defining the perimeter of the section (input).

(2) NI: The number of points defining the section (input).

(3) CCD: The diameter of the circumscribing circle (output).

(4) XC,YC: The coordinates of the center of the circumscribing circle (output).

(5) I1,I2,I3: The indices to the points in arrays X,Y which define the circumscribing circle (output). If the circle is defined by two points, I3 will be set to zero.

and track the light pen, and ADDPNT is used to insert the indicated perimeter point into the die polygon data arrays. It should be noted that the point added must lie between the cave axis points or it will not be ACCEPT'ed.

After adding a new point, or skipping this operation, as desired by the user, all points between the cave axis end points are shifted. This is done by determining the distance W along the axis from an intermediate point to the closest axis end point. The distance W times the cave compensation factor (inches/inch) specified by the user results in the amount that the point will be shifted along the normal. This would result in point C in Figure V-6 being moved to C'.

When all points between the axis end points have been compensated, the die polygon points are redrawn. Because the amount of shift of the points is usually small, it is very difficult to visually see any difference in the location of the shifted points. A comparison of the numeric data, however, will show that the cave compensation has been made.

Subroutine DIEHOLE

DIEHOLE determines the minimum number of openings in a die. This value, along with the maximum set by the press characteristics, provides the limits for subsequent computations for the number of openings.

The user is first queried as to the average bearing length to use for load calculations. A default value is provided for the user if he wishes to use it. Then, starting at N=1, the load for increasing number of die openings is calculated. The load will decrease with an increasing number of openings, due to the decrease in extrusion ratio or die resistance. When a load is found which is less than the press capacity, the number of openings at this point becomes the lower limit when the number of openings is specified or calculated. If the number of openings required in order to have the press load less than the press capacity is greater than the maximum number allowed for the press, the error flag is set and DIEHOLE is terminated.

When the minimum number of openings is found, this value and the associated load is reported. The user is then requested to enter the minimum extrusion length. This is subject to the following condition:

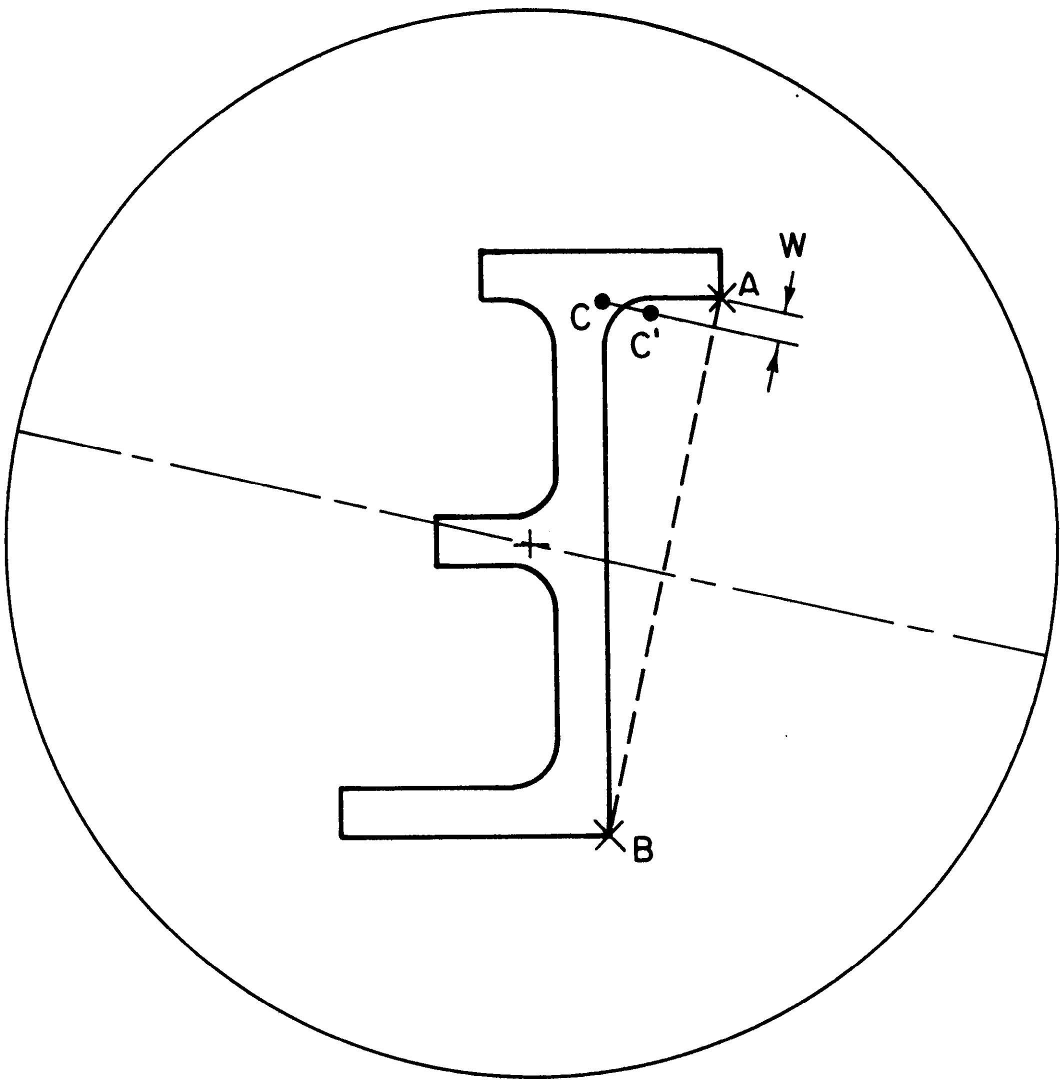


FIGURE V-6. DIE CAVE COMPENSATION GEOMETRY described in detail in Appendix I. Essentially, the number of openings can be set by user directive, or analytically based on maximum length, maximum billet yield, or maximum production rate.

This subroutine is contained in module OVRL1E.

Subroutine DISTRS

DISTRS determines the stresses resulting from (1) the extrusion load acting on the bridge area between openings in a multi-hole die, (2) the cantilever beam effect of a tongue. The theory underlying these analyses is given in Appendix I. The geometry for the die bridge stress calculations is shown in Figure V-7. The area defined by points 1, 2, 3, . . . 19, 20 has a load imposed on it by the extrusion process. This load must be supported by areas defined by lengths (1,2) and 10,11) times the die thickness.

When called, DISTRS first determines the average extrusion pressure on the die. It then calls TOOLCH to obtain the tool thicknesses and tool-to-tool clearances. If the die has multiple openings and the user has specified that a stress analysis is to be made, GETPTS is used to obtain the coordinates of the bridge area and the shear boundaries. These coordinates are scaled to user units, and the area of the bridge and the resulting load are determined.

The shear lengths for each tool are found by summing the distance between each pair of shear boundary points. The backer and bolster have their shear lengths reduced by twice the clearance between this respective tool and the one in front of it. If a shear length of less than zero is calculated, the shear length is set to zero. The shear area for each tool is finally determined as the shear length for each tool times its respective thickness. The stress in the three combinations of (1) die only, (2) die and backer, no bolster, and (3) die, backer and bolster is found by dividing the load on the bridge by the appropriate sum of shear areas. These results are sent to both the CRT and line printer file. The user may then repeat the calculations for some other bridge area and shear boundaries.

The result of the bridge stress analysis are not used directly in the subsequent design of the die. Rather, they are meant as indicators to the designer of the degree to which conforming or non-conforming support tools may be needed.

$$L_m + \text{Loss} \leq L_{min} \leq L_{Ro} - \text{Loss} \qquad \text{(V-8)}$$

where

L_m = desired length multiple

L_{min} = minimum extrusion length

L_{Ro} = runout table length

Loss = breakthrough and stretcher loss length. (Set to 8 feet by a data statement in ALEXTR)

The maximum number of die openings is next calculated. This is the number of openings which will produce pieces of the minimum specified length. If this value for the maximum number of openings is greater than the number of openings specified by the press characteristics, the latter becomes the maximum used in subsequent calculations. However, if whatever value determined for the maximum number of openings is less than the minimum number of openings required to meet the press capacity, an error condition exists.

This process can be illustrated by an example. Assume the following:

(1) Maximum number of openings specified by press characteristics: 6

(2) Number of openings to produce minimum length extrusions: 4.

In this case, the number of openings based on volumetric condiderations will be set to 4. If, however, in order to have the expected load less than the press capacity, it is necessary to have at least 5 openings in the die, the part cannot be made in the press configuration specified. If a 5-hole die was used, none of the extrusions would be long enough to be of any value. If a 4-hole die was used, the press could not generate sufficient load to extrude the sections.

When the upper and lower limits for the number of die openings is established, DIHOLE queries the user as to the method to use to actually set the number of openings. The techniques and analysis behind this operation are

After completing the bridge stress analysis, DISTRS determines the load and stress on die tongues. A tongue is a portion of a die which is subject to loads similar to a cantilever beam. The area 2, 3, 4, 5 in Figure V-8a is an example of a tongue.This tongue is subject to bending and shear stresses about the axis defined by line 2, 5. The bending is shown, exaggerated, in Figure V-8b.

To find the tongue stresses, the user is first requested to identify the two points defining the tongue, i.e., points 2 and 5 for the tongue shown in Figure V-8a. This is done by using GETHIT. This returns the indices to the points selected. The area bounded by the tongue points identified is found by using AREAX and the width of the tongue (the throat opening) is determined. TNGSTR is then used to determine the stresses in the tools. This evaluation is made for three conditions:

- Die, backer and bolster tongues
- Die and backer tongues
- Die tongue only.

After printing the results of the stress calculation, TNGCMP is used to shift the extreme points of the tongue back towards the tongue bending axis. The shifted tongue points are displayed on the CRT along with the original die polygon points. Usually, however, the difference in position of the shifted points is too small to detect visually. The user may then identify additional tongues or terminate DISTRS, as desired.

DISTR is contained in module OVRL1J.

Function DISTSQ (X1,X2,Y1,Y2)

DISTSQ returns the square of the distance between two points. That is:

$$DISTSQ = (X1 - X2)^2 + (Y1 - Y2)^2 \qquad (V-9)$$

DISTSQ is part of the root segment module ALEXTR.

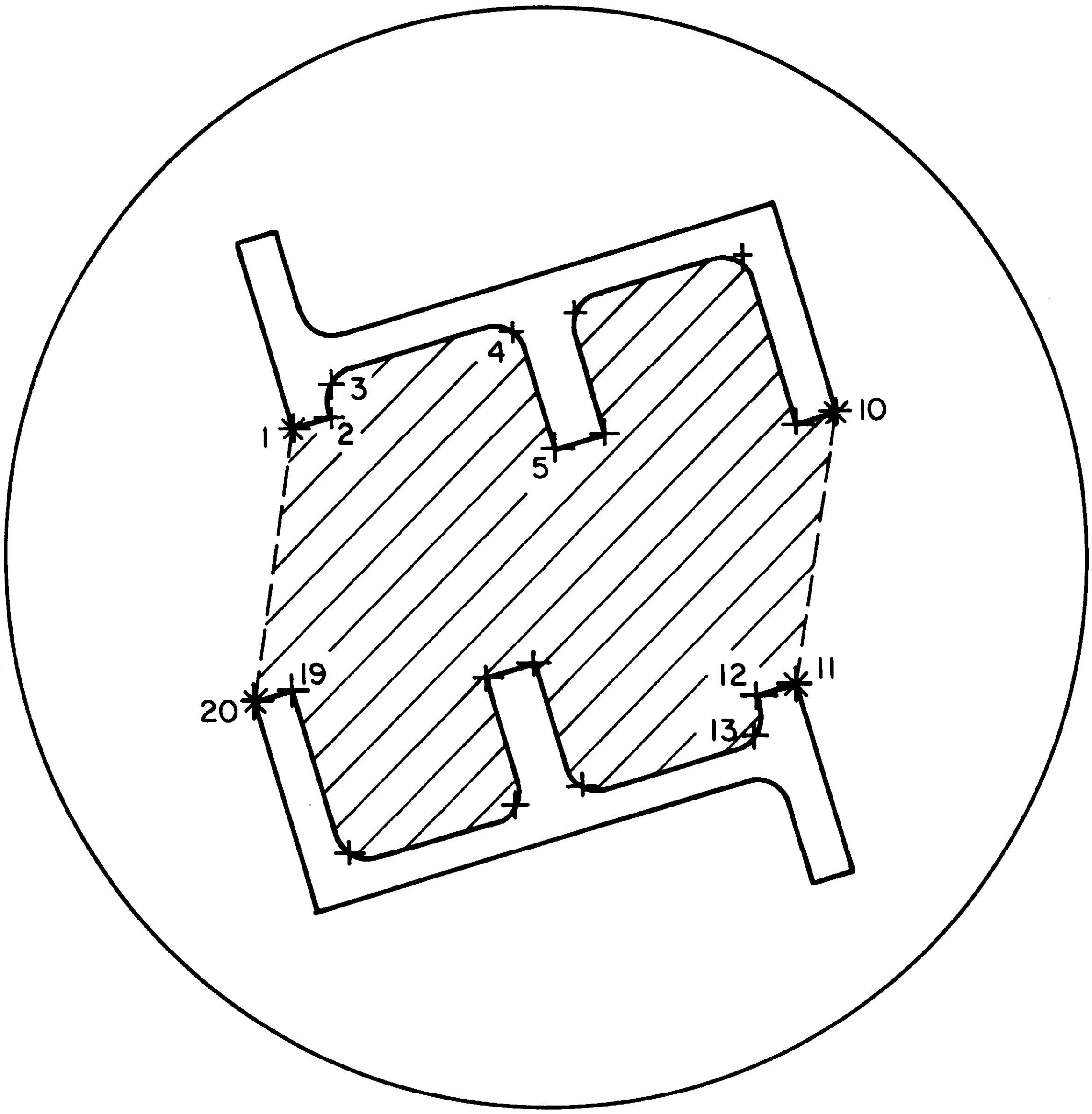


FIGURE V-7. GEOMETRY USED FOR DIE BRIDGE STRESS ANALYSIS

Subroutine DRWCRC(R,K)

DRWCRC draws a circle at intensity K with a radius R. R is prescaled in raster units. The circle is generated as a series of incremental vectors. The number of vectors used to draw the circle is proportional to the size of the circle. The center of the circle is the coordinate pair IXCTR,IYCTR contained in Common Block DISPLA.

DRWCRC is contained in module OVRL2D.

Subroutine DRWSEG(ALF,CGSEG)

DRWSEG draws the pie shape on the CRT which represents the die segment used for one opening of a multi-hole die. The display is generated by positioning the beam to the bottom center of the CRT using absolute coordinates and LINESG. The beam is then incrementally moved to the die perimeter, counterclockwise around the die the appropriate amount and then back to the center. The number of increments used to define the arc of the die varies inversely with the number of openings. That is

$$N_I = 120/N_O \tag{V-10}$$

where

N_I = number of angular increments

N_O = number of openings.

The value 120 was experimentally determined as a value which yielded a reasonably smooth arc while minimizing the number of graphic elements.

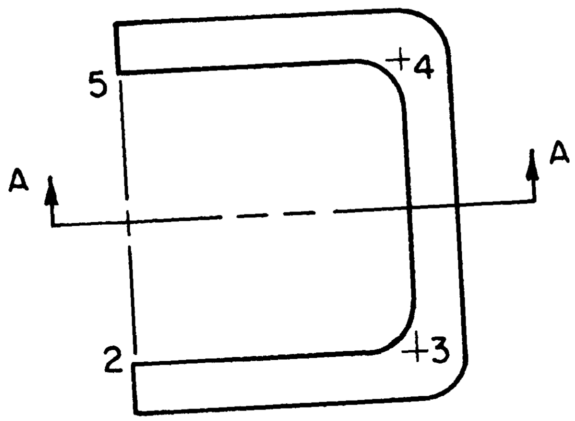
After the segment is drawn, its CG is marked with a "+".

The variables in the calling sequence are

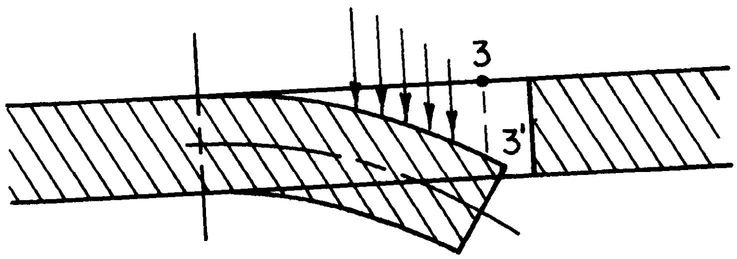
(1) ALF: The included angle of the segment.

(2) CGSEG: The center of gravity of the segment in the user's units relative to the center.

DRWSEG is part of OVRL2D.



(a) Tongue Geometry



(b) Tongue Deflection Under Load

FIGURE V-8. TONGUE STRESS ANALYSIS AND DEFLECTION COMPENSATION

Subroutine FITARC

(X,Y,R,XP,YP,NP,RES,LIM,IFLAG)

FITARC calculates the coordinates of line segments which can represent a radius to a given resolution. A complete description of the technique used is contained in Appendix II.

The variables in the calling sequence are as follows:

(1) X,Y,R: Coordinate arrays defining a corner or fillet (input).
(2) XP and YP: X and Y coordinates of the newly defined line segments (output).
(3) NP: the total number of points calculated for a section (output).
(4) RES: the arc distance between consecutive points on the radius (input).
(5) LIM: the maximum number of points which can be calculated for a section (input).
(6) IFLAG: a flag indicating that NP equals LIM (output).

FITARC is part of module OVRL2A.

Subroutine FNDCRC

(X,Y,N,JL,J2,J3,MARK,XC,YC,RTC)

Given a set of three points, FNDCRC determines the center and radius of the smallest circle which will circumscribe the points. This circle does not necessarily pass through all three points. A computer description of the algorithms used is contained in Appendix II. In summary, the subroutine does the following:

- Determine if triangle is acute or obtuse based on relative size of sides.

Subroutine ENDFLS

ENDFLS closes scratch disk files. These files were created in PLACEM and MODYLO when going through a die layout process. ENDFLS is called when these files are no longer needed.

ENDFLS is part of module OVRL2G.

Subroutine ENDIT

ENDIT does housekeeping functions to terminate execution in an orderly fashion when the user wants to cease use of ALEXTR. It performs the following functions:

(1) Clear the CRT and terminate the graphic operating mode.

(2) Mark the end of all files created during the execution of ALEXTR and close all files used.

ENDIT is the last subroutine called before ALEXTR terminates. ENDIT is part of module OVRL1D.

Subroutine ERRPRT

This subroutine formats and outputs messages describing a variety of error conditions. Most of these messages have to do with problems encountered while trying to read data for a section, or when determining the number of openings in a die. The specific message generated is determined by the value of IERROR. IERROR is a variable in Common block SECTN. The message is output to both the CRT and printer.

ERRPRT is part of module OVRL1C.

with the L.P., and the coordinates of the hit. If the first hit is on the "ACCEPT' text, the program loops and looks for another hit, since an item cannot be accepted until it is identified. If the hit was on a display item, the dummy point is deleted and replaced with the coordinates of an X to indicate where the hit was made. The item tag returned from the valid L.P. hit, less L, is saved in array K.

Another L.P. hit is then looked for. If the second hit is on the "ACCEPT" text, the item previously hit is desensitized so it will no longer respond to L.P. hits. The value "L" is then subtracted from the tag value previously saved. This allows the indices returned to represent the actual array indices for the point, rather than the display item index. The Do-loop then continues or terminates, as appropriate.

If the second hit was not on "ACCEPT", but rather on another L.P. sensitive display item, the program loops back. The display item for the cross at the previous position is deleted and replaced with a cross at the current location. Once the Do-loop is satisfied, the "ACCEPT" text is blanked before the subroutine returns control to the calling program.

Data is passed via Labeled Common block DISPLY, and the following parameters in the calling sequence:

(1) K: the name of the array used to store the modified display item tag values (input and output).

(2) J: the number of L.P. hits to be obtained (input).

(3) L: the offset between display items and the associated element of the coordinate arrays (input).

(4) M: a flag. If M = 0, the item hit is made to blink until ACCEPT or another item is hit.

GETHIT is part of module OVRL2E.

Subroutine GETPTS

(NOPENS,IX,IY,N1,N2)

GETPTS returns two sets of X,Y coordinates. The first set defines the bridge area between two or more die openings. The second list contains the

- If obtuse, the circle is defined by the end points of the longest line.
- If acute, use equation of circle to develop simultaneous equations. Solve these for definition of circle.

The arguments in the calling sequence are:

(1) X,Y: A set of coordinate points defining the perimeter of a section.
(2) N: The number of coordinate pairs in X and Y.
(3) J1,J2,J3: The indices to the three points in X, Y for which the circumscribing circle is to be found.
(4) MARK: A flag. As an input, if MARK = 4, the indices J1,J2,J3 are interchanged as necessary so that J1 and J2 define the end points of the longest of the three lines. As an output:
   - If MARK = 2, CC is defined by two points.
   - If MARK = 3, CC is defined by three points.
(5) XC,YC: The center of the circumscribing circle.
(6) RTC: The radius squared of the CC.

FNDCRD is part of module OVRL2B.

Subroutine GETHIT

(K,J,L,M)

Subroutine GETHIT obtains light-pen (L.P.) hits on sensitized display items, and returns the tag numbers for each item hit.

When first called, GETHIT restores the "ACCEPT" text as an L.P. sensitive item. It then starts a Do-loop to get L.P. hits on J items. J number of items must be picked in order for this routine to terminate. After starting the loop, a dummy, nondisplayed point is created. This will be replaced when an item is actually hit, and an X is drawn to indicate the position of the hit. A call to LGTPEN is then issued which returns the item number of the entity hit

Module GRPHCS

This module consists of a number of FORTRAN callable subroutines which allow the user to create and manipulate graphic images and text on the GT40 display system CRT. All coordinates are in screen units. X axis has a range of Ø - 1Ø24, y axis Ø - 768. GRPHCS is linked with the root segment, ALEXTR.

CALL BLANKF(DFILE)

Blanks a display file.

CALL BLANKI(TAG)

Blanks the item identified by its tag.

CALL CHNGIT(ITAG,INT,LPNS,BLNK,TYPE,ITALCS)

Changes the attributes of an item identified by its tag. All attributes are optional, and may be skipped by inserting a blank between corresponding commas. See LINESG for description of attributes.

CALL CLEAR

Clears and reinitializes the display file handler. DSPLYG must be re-called after CLEAR.

CALL DSPLYG(DFILE,L,ITEMS)

This call establishes a buffer area for the graphics file items to be created by subsequent calls to LINESG, POINTG, etc.

DFILE: Display file area; a one-dimensional array in the calling program.

L: Length of the array.

ITEMS: Number of items one expects to store in this Display File. Each call to LINESG,POINTG, etc., creates one item.

CALL EXITG

To terminate Graphics Mode.

CALL GTPSTN(ITAG,IX,IY)

Returns the present starting position of item ITAG.

CALL LEGNDG(ITAG,IX,IY,N,CHAR,INT,LPNS,BLNK,ITALCS)

Displays a character string starting at IX,IY. The arguments INT, LPNS,BLNK,ITALCS are optional.

coordinates which define the bridge openings or shear boundaries. Figure V-7 shows a bridge area, cross-hatched, defined by points 1, 2, 3, . . . 19, 20. This area, B_A, is subject to pressure $\overline{P}$ during extrusion. This results in a shear stress, τ, across the bridge boundaries AB and CD. The bridge boundaries must be strong enough to withstand the shear stress or tool failure will occur.

When called, GETPTS first makes the display item for each opening light pen sensitive. The operator then uses the L.P. to locate each bridge area point. The user may start at any point and go either clockwise or counter-clockwise. Once started in a particular direction, however, he must continue in this direction, locating each point in its correct sequence. As each point is located, it is marked with a "+". To fix the position of a located point, the "ACCEPT" L. P. is used. Once all points are located, the "END" light button is used to indicate that GETPTS is to proceed to the next operation.

When "END" is hit, the openings are made non-L.P. sensitive. At the same time, the "+" marks indicating the bridge areas are changed to be L.P. sensitive. The user is then instructed to indicate which of the points defining the bridge area also define the bridge boundaries. Using the L.P., the user picks the bridge boundary points. These points are marked with an "X". The bridge boundaries in Figure V-7 are points 1, 20, 10 and 11. The boundary points must be indicated as sets of pairs. That is, points 1 and 20 must be indicated as a pair, and 10 and 11 as a pair. The sequence of the points within the pair is immaterial. Points 1 and 20 may be picked as (1,20) or (20,1).

When the location of shear boundaries is completed, all items are de-sensitized and the light buttons are blanked. The shear boundaries are then displayed as dashed lines.

The arguments in the calling sequence are:

1. NOPENS: The number of openings in the die
2. IX,IY: Coordinate pairs in raster units
   The total number of pairs is N1 + N2.
3. N1: The first set of X,Y pairs. N1 is the number of coordinates which define the bridge area
4. N2: The second set of X,Y pairs, N2 is the number of coordinate pairs which define the bridge (shear) boundaries.

GETPTS is part of module OVRL2G.

ITAG: The tag assigned to the item created by this call.

N: Number of points to be displayed.

IX,IY: Arrays containing the coordinates.

INT: Intensity Ø - 7.

LPNS: Light pen sensitivity.

BLNK: Blinking status.

CALL PRNTCH(IC)

Print a character on the screen and/or the Decwriter.

CALL READCH(IC)

Get a character from the keyboard. If no character is available, IC = Ø.

CALL REMOVF(DFILE)

To purge a display file.

CALL REMOVI(ITAG)

Removes all items with tags equal to and greater than ITAG from the display file, reclaiming memory space. If ITAG is greater than any existing item identifier, message "UNASSIGNED TAG" is returned. Upon return, ITAG is one less than when called.

CALL RESTRF(DFILE)

To restore a blanked display file.

CALL RESTRI(ITAG)

Restores a blanked item.

CALL SCROLG(NLINES,IYTOP)

To adjust scroller parameters. Graphics Monitor must be in use.

NLINES: Number of lines to be displayed.

IYTOP: Y-coordinate of the top line. Each line is 25 units vertical.

CALL VECTOR(IDX,IDY,K)

An incremental vector is added to the most recent display item. This item would have been generated by a call to LINESG. If X = Ø, the new vector will not be displayed; if K = 1, the new vector will be visible.

N: Number of characters to be displayed

CHAR: An array containing the characters

INT: Intensity

LPNS: Light pen sensitivity

BLNK: Blinking status

ITALCS: Italics front or normal.

CALL LGTPEN(ITAG,IX,IY)

To get the tag and coordinates of the first light pen hit after the call.

CALL LINESG(ITAG,N,IX,IY,INT,LPNS,BLNK,TYPE)

Displays a series of vectors connecting the input coordinates. The agruments INT,LPNS,BLNK and TYPE are optional.

ITAG: Is the identifier assigned to the item created by the call to LINESG.

N: Number of coordinate pairs.

IX,IY: Coordinate pairs in object space units.

INT: Intensity at which the item would be displayed
Ø = dimmest, 7 = brightest.

LPNS: Light pen sensitivity Ø = not sensitive, 1 = sensitive.

BLNK: Blinking status, Ø steady, 1 = blinking.

TYPE: Line type; Ø = solid
1 = long dash
2 = short dash
3 = dot dash.

CALL LINKRT

Links the display file handler to the RT-11 operating system.

CALL MOVEIT(ITAG,IX,IY)

Item ITAG is moved to absolute position IX,IY.

CALL NEXTAG(ITAG)

The next available tag will be returned in ITAG.

CALL POINTG(ITAG,N,IX,IY,INT,LPNS,BLNK)

Displays a series of points. The arguments INT,LPNS,BLNK are optional. Any of them may be omitted by inserting a blank in between the corresponding commas.

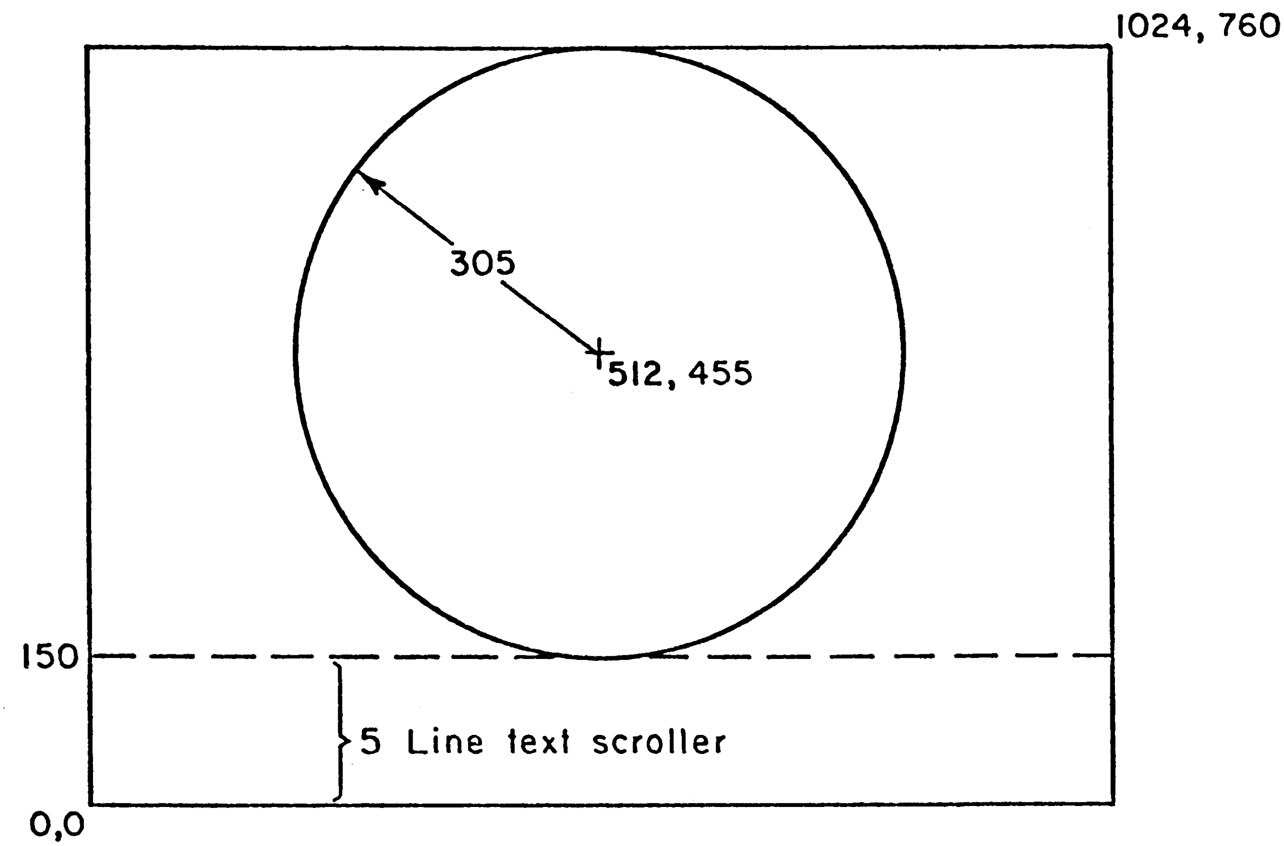


FIGURE V-9. USE OF CRT. DIMENSIONS SHOWN ARE RASTER UNITS

Subroutine INITDS

INITDS initializes the graphics display software each time a new section is processed. This operates with CRT coordinates as shown in Figure V-9. A five line area is established across the bottom of the CRT for text. Text written to the CRT will scroll off the top of this area as new lines are added. A circle is generated at the center of the area above the scroll space. This circle is used to represent the circumscribing circle or the extrusion press container, depending on the situation. The scale factor is established as

SCALEF = 610/C rasters/inch.

The displayed circle has a diameter of 305 raster units and C is the diameter in the user's units of the circle to be displayed. The circle itself is drawn using DRWCRC.

INITDS also generates two light buttons along the left-hand side of the CRT. These are the words "END" and "ACCEPT". They are created and then immediately blanked from view. They are restored to view, as needed, in other subroutines.

This subroutine is contained in module OVRL1B.

Subroutine INTRPL
(XI,YI,NI,X,Y,R,NP)

Given a shape defined by an array of fillet or corner coordinates and the radius associated with each, INTRPL will calculate the array of points needed to describe the radii by short-line segments. The first array of points (polygonal shape) is referred to as the part file. The second array of points (smoothed shape) is referred to as the display file. It is this latter shape which is displayed on the CRT, after being scaled and translated as necessary.

The routine starts by initializing the arc length to zero, the count of points in the display file to 1, and setting the first point in the display file equal to the first point in the part file. The minimum segment size is added to the arc length to establish its initial value.

Subroutine LINEQ
(X1, Y1, X2, Y2, COEF)

Given the coordinates of two points, X1, Y1 and X2, Y2, LINEQ determines the coefficients of the straight-line defined by the points. The coefficients are those for the general form of the equation of a straight-line which is A*X + B*Y = C. A, B, C are returned as the elements of the three element array, COEF.

LINEQ is part of module OVRL2H.

Subroutine LOAD(N,TOTLOD)

LOAD calculates the load required to start a section extruding using a direct extrusion process. The details of the load calculations are given in Appendix I.

The parameters in the calling sequence are:

(1) N: The number of openings in the die (input)

(2) TOTLOD: The expected total load required to start the extrusion (output).

LOAD is part of module OVRL2C.

Subroutine LYONEW

LYONEW is called from PLACEM if PLACEM was called directly after completing a stress analysis. Such an instance might occur when a stress analysis indicates a preliminary die layout results in unworkable stresses and the user wishes to modify the layout. LYONEW asks the user: "Continue with existing layout (Y/N)?". If the user types anything but Y(yes), subroutine ENDFLS is called and then control is returned to PLACEM.

If the user types Y(yes), the scale factor for the full die layout is calculated. The mirror image parameter for the first hole is also reversed.

A DO-loop is then started which generates the additional points needed to define the radii. A test is made of the absolute size of each radius. If it is less than the current arc length, it is considered too small to bother interpolating. Instead, the point in the part file defining the location of the radius is equated to the point in the display file. After each radius is interpolated, the number of points in the display file is compared to the maximum number allowable. If they are equal, the arc length is increased and the entire procedure restarted.

If the radius is large enough to require fitting of line segments, the interpolator subroutine FITARC is called. This operates by passing to it three points defining two lines, the radius at the intersection of the two lines, the name of the array where the interpolated points will be store, the arc length to be used, and the maximum allowable number of points in the display file. The routine returns the interpolated points (values and count) and a flag indicating if an attempt was made to calculate more points than would fit in the output array. This flag is tested and if it is set, a branch is made to the beginning of the program where the arc length is increased. FITARC is called until all sets of three adjacent points have been interpolated.

The variables in the calling sequence are:

(1) XI and YI: Arrays of the coordinates of the end points of line segments needed to display a curved figure as a series of straight-lines (output).

(2) NI: The number of elements in the above arrays; maximum value = 500 (output).

(3) X and Y: Arrays of coordinates defining a polygonal shape (input).

(4) R: The array defining the radii at each vertex of the polygon (input).

(5) NP: The number of elements in the arrays XPS, ZPS, and RPS (input).

INTRPL is part of module OVRL2A.

Subroutine MARKXX(I,J)

MARKXX draws an "X" on the CRT at screen coordinates (I,J). It is used and operates in a manner very similar to MARKIT. MARKXX resides in the root segment as part of module ALEXTR.

Subroutine MATERL(FS,SPEED)

MATERL allows the user to indicate the aluminum alloy to be used for the extrusion. When an alloy is selected, the flow stress and nominal extrusion speed are found from a set of tables.

When first called, MATERL blanks from view all items currently displayed on the CRT. The alloy designators, such as 1050, 1100, 7075, etc., are then displayed as a vertical column of light buttons. The "ACCEPT" light button is also restored to view and the user is instructed to pick and accept an alloy. The light pen is used to select the desired alloy. As an alloy designator LB is touched with the LP, the LB will start to blink. If another alloy LB is then touched, the previous blinking item will stop blinking and the latest hit will start blinking.

When the user has picked the desired alloy, he must then "ACCEPT" it to have the selection registered. Picking "ACCEPT" without first having selected an alloy will have no effect. When an alloy pick is registered, the alloy LB's are erased, the "ACCEPT" LB is blanked and the previous display items restored to view. Before returning, MATERL determines the flow stress, average extrusion speed, average extrusion temperature, and the coefficient of the thermal expansion for the selected alloy from a set of tables.

The parameters in the calling sequence are:

(1) FS: The flow stress for the alloy selected (output).

(2) SPEED: The typical extrusion speed for the alloy selected (output).

MATERL is part of module OVRL2B.

This allows the die layout to be generated by specifying a second mirror image operation via ROTMIR on the first opening when control is returned to PLACEM.

LYONEW is part of module OVRL2G.

Subroutine MARKCG(N)

MARKCG marks with a '+', the center of gravity of each segment, for the display of a multi-opening die. The position of the first CG is located on the upwards vertical relative to the center of the die. This position is then rotated by the included angle of a single segment for all remaining segments. The actual drawing of the '+' is done by MARKIT. N is the number of openings in the die.

MARKCG is part of module OVRL2D.

Subroutine MARKIT(I,J)

MARKIT draws a cross (+) on the CRT at screen coordinates (I,J). It is used to mark specific points of interest such as center of a circle, or the center of gravity of a section. The routine first positions the CRT to the point I,J by using LINESG. This is followed by calls to VECTOR to draw the following elements:

(1) An invisible vector drawn upward, 10 units long

(2) A visible vector drawn down, 20 units long

(3) An invisible diagonal vector, 10 units up and 10 units to the left

(4) A visible horizontal vector, 20 units to the right.

This program element resides in the root segment as part of module ALEXTR.

relative to the center of the container or die. The user is then asked to indicate what change he wishes to make by typing the appropriate response to the question:

TRANSLATE(1), ROTATE(2), MIRROR(3), TEST(4), DONE(5)?

The user's response is tested to see that it lies in the range of 1 to 5. If it does not, the question is repeated. The display item tag is then determined based on whether the item to be changed is a single hole die, a single segment of a multi-hole die, or the full display of a multi-hole die. If the item is to be translated in X and/or Y, TRNSLT is called to make this change. If rotation or mirror image is to be done, ROTMIR is called and passed the proper parameter for the change desired. If TEST is requested, the tolerance is requested and then the scaled, light pen sensitive tolerance circle is drawn. When DONE is specified, the program jumps back to the beginning and re-asks if the opening position is OK.

When working with a multi-opening die, the procedure for modifying the single segment display is similar to that described above. However, once the user approves the position of the single segment opening, the single segment display is erased and replaced with a display showing the container with as many segments and openings as were determined in DIHOLE. The translation, rotation and mirror image parameters are computed based on the parameters for the first segment using ALLSEG. This also generates the display images of the segment dividers. The opening is then scaled relative to the full size container and the opening display file is generated and saved on disk. For each opening, this file is then mirrored (if this quality was previously specified), rotated, and translated as appropriate. Finally, each individual opening and its CG is displayed using LINESG and MARKXX.

It should be noted that because the individual openings are drawn in sequential order and new display items cannot be inserted in the middle of a display file, the entire process of creating a new display file for each item must be done whenever a rotation or mirror image change is specified. Translation changes can be made by using MOVEIT to change the starting position of a unique display item. Although re-drawing all openings after each change sounds

Subroutine MAXMIN

(A,AMAX,AMIN,IMAX,IMIN,N)

This subroutine determines the maximum and minimum values of an array, and their respective indices. It first initializes the maximum and minimum values to the first element of the array, and the indices to 1. It then tests all other values (from 2 to N) to see if the test value is a new minimum. If it is, it saves the current value being tested as the minimum, and the index of the value being tested. It then performs a similar operation to test for a new maximum.

Variables in calling sequence:

(1) A one-dimensional array of real values (input).
(2) AMAX: Maximum value found in the array (output).
(3) AMIN: Minimum value found in the array (output).
(4) IMAX: Index to the maximum value in the array (output).
(5) IMIN: Index to the minimum value in the array (output).
(6) N: Number of elements in the array to be tested (input).

MAXMIN is part of module OVRL2A.

Subroutine MODLYO

MODLYO allows the user to interactively modify the position of an opening on a die. For multi-hole dies, the position of the first opening may be modified and then repeated for all other segments. Or the positions of the individual openings may be modified as desired, one at a time.

When first called, PLOTIT is called and offers to plot the die segment and opening as located by PLACEM. This is followed by the query "POSITION OK (Y/N)?". If the user responds Y(yes), MODLYO terminates.

If the position of the opening is not acceptable and the die has a single opening, the current parameters for CG position, rotation and mirror image are typed. The CG position is the position of the CG of the opening

The interactive modification of opening placements in a multi-hole die may be continued as long as desired. The process may be stopped by selecting the "END" light button, or specifying the "DONE" parameter once an item is picked. When the modification process is terminated, the user is offered a hard copy plot. This is followed by the query "POSITION OK?". For a multi-hole die, MODLYO will not terminate until the multi-hole display has been generated and this question has been answered with an affirmative Y(yes). For a multi-hole die, this question must be answered with Y twice: once for the single segment display, and the second time for the full, multi-opening display.

This subroutine is contained in module OVRLIG.

Subroutine PARAMS

PARAMS formats and outputs the values calculated for area, perimeter, shape factor, centroid and the center and diameter of the circumscribing circle for an extrusion section. The shape factor is calculated as the ratio of the perimeter to the weight per foot. The results are sent to both the CRT and printer file. The values output by PARAMS are passed to it via named Common SECTN.

PARAMS is contained in module OVERL1A.

Subroutine PICTUR

This subroutine is used to initialize the graphics display software and draw the extrusion section and its circumscribing circle. The graphics software is initialized by calling subroutine INITDS. SCALZZ is then used to generate the section display file in absolute CRT raster units. This is followed by a call to LINESG to actually display the section image. PLOTIT is called to provide a hard copy plot, if desired, of the CC, the center of the CC, and the extrusion section. These are display items 3, 4, and 5, respectively.

PICTUR is contained in module OVRL1B.

time-consuming, experience has shown it to be surprisingly fast with the time required not being sufficient to annoy the user.

Because the number of items displayed varies with the number of openings, PLTSEG is used to generate a file of display items which may be plotted. PLOTIT is called to determine if a hard copy plot is actually desired.

Once the multi-hole layout is displayed, the user is given the opportunity to modify the position and orientation of one opening without disturbing the others. To do this, the user must identify which of the openings is to be changed. This is done by using the light pen. The displays of each opening are made light pen sensitive and the "END" and "ACCEPT" light buttons are restored to view. As each opening is touched with the L.P., the opening will start to blink. If one opening is blinking and another opening is touched, the first will stop blinking and the second will start blinking. This will continue until a light button is hit. If "END" is selected, the program loops and asks if a hard copy is desired. If "ACCEPT" is hit, the last opening selected will stop blinking but its CG "X" marker will blink instead. The appropriate index for the item picked is determined and the translation, rotation and mirror image parameter values are typed. This is followed by the request to indicate what modification is to be made to the item selected, in the same manner as was described above. The requested change is made and the display regenerated to show the effect of the latest change.

It should be noted that when the translation and rotation values are typed, they are in absolute terms. That is, they reflect the sum of all changes made to get the opening to its current position, relative to its original polygon input definition. When change values are requested for these two parameters, however, the values entered are incremental amounts. That is, entering 45 as a rotation angle will cause the opening to be rotated an additional 45 degrees from whatever the current position is.

The mirror image parameter has the value +1 if not mirrored, -1 if mirrored. Whenever a change to the mirror image is requested, the current value is negated. If the result of the negation is +1, the mirror image transform process is skipped.

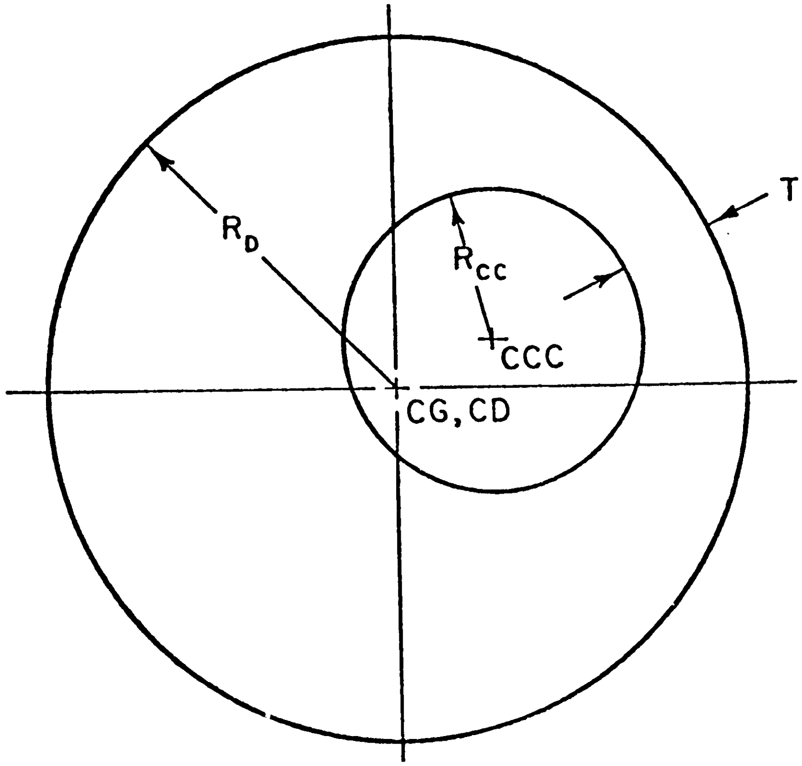


FIGURE V-10. ORIGINAL LOCATION OF AN OPENING RELATIVE TO THE DIE CENTER

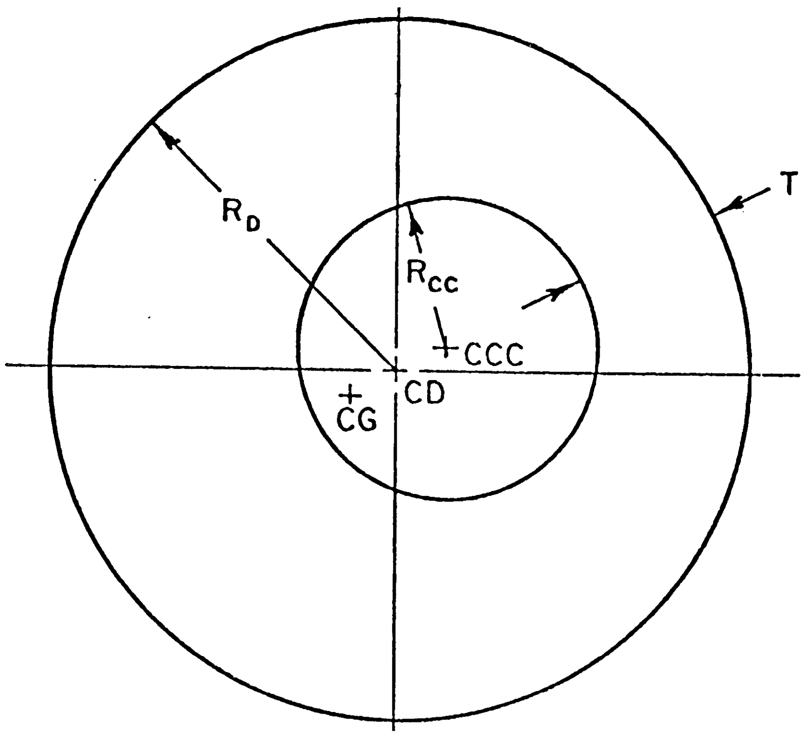


FIGURE V-11. OPENING TO MEET CLEARANCE AFTER SHIFTING THE CENTER OF GRAVITY OF THE CRITERIA

Subroutine PLACEM

PLACEM is the routine which locates each opening on the die. For a die with one opening, the CG of the opening is initially placed at the center of the die (CD). The clearance between every point on the opening and the container diameter is then found. If the clearance at all points does not equal at least .75 inch, the center of the circumscribing circle (CCC) is shifted toward the CD by half the current difference between the two. An earlier test verified that the circumscribing circle would fit the container. Thus, by using half the difference each time, the CCC will quickly converge to the CD, if necessary, to meet the clearance condition. This process is illustrated in Figures V-10 and V-11. In Figure V-10, the CG of the opening is located at the center of the die. The minimum distance between the opening and the container could be distance T. If T is less than the desired clearance value, the CCC would be moved toward the CD so that the minimum distance becomes T. This is again checked against the tolerance specified and the position accepted or changed again as appropriate.

When the position of the opening has been determined, the display of the opening on the die is generated. The scale factor is calculated and then subroutines SCALZZ, LINESG, and MARKXX are used to scale and display the image. The display image file is stored as a scratch file on disk. As subsequent user-directed changes in the orientation or position of the opening are requested, the changes are made by applying the appropriate transform to the scaled data in this file. This eliminates the need to re-scale the data each time.

Before exiting, the parameters identifying the location of the opening on the die are recorded. These parameters are:

(1) DXCG, DYCG: The position of the opening CG relative to the die CG. This would normally be expected to be Ø, Ø for a single hole die.

(2) THETA: The amount the opening has been rotated, relative to the original section definition. This is zero for the automatic, single-opening placement. This angle is in degrees.

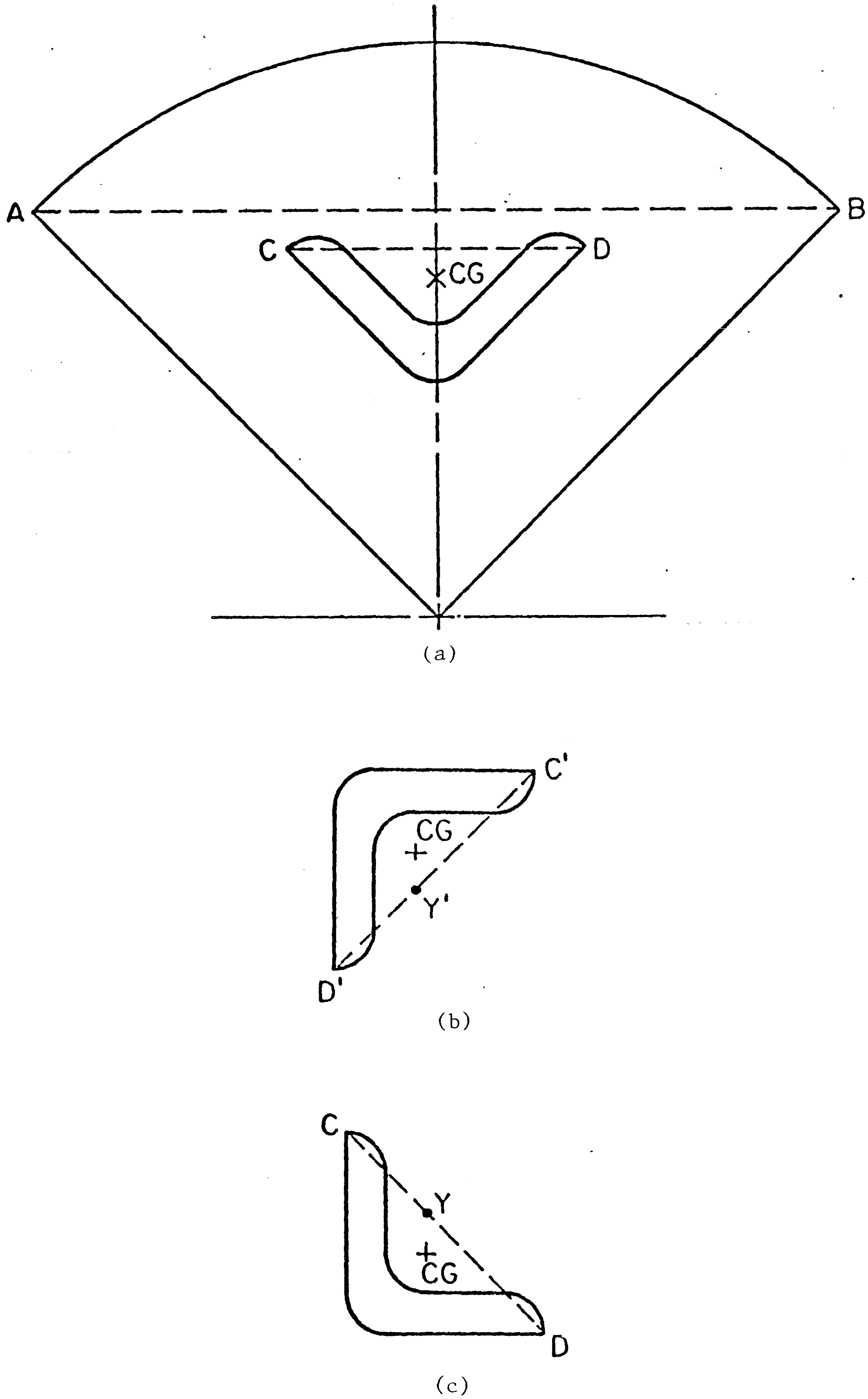


FIGURE V-12. LOCATION AND ORIENTATION OF ORIFICE IN A MULTI-HOLE DIE

(3) MIRROR: Would be set to 1, indicating a mirror image transforms was not made.

If a multi-opening die is being designed, the included angle of the segment and its CG are first determined as shown in Figure V-12a. The greatest dimension between any two points on the section perimeter is found next. This is done by computing the distance between every pair of points defining the input polygon and comparing the value to the previous maximum.

Once the maximum part dimension is found, it is compared to the chord of the segment (Line AB, Figure V-12a). If the greatest part dimension is larger than the chord, it is assumed the part will not fit the segment and an error return is made. If the greatest part dimension is less than the chord, it is presumed the part will fit the segment. The angle of the greatest dimension and its position relative to the CG of the section is then determined. The objective is to rotate the section such that the greatest dimension of the part is parallel to and as close as possible to the chord of the segment. The relationship of the CG to the greatest dimension is found by determining the Y-intercept at the intersection of a vertical line through the CG and the greatest dimension line. This is done using subroutine XYINTR. This is shown in Figure V-12b and V-12c. It is necessary to rotate the section shown in Figure V-12b by -45 degrees to obtain the arrangement shown in Figure V-12a. The section shown in Figure V-12c would be rotated by 225 degrees to obtain Figure V-12a.

Additional consideration is given when there are two openings. In this case, the segments are semi-circles and the chords are the horizontal diameters. In order that the longest part dimension be parallel and as near to the chord as possible, an additional 180 degrees must be added to the rotation value.

When the rotation angle is determined, the section display file is generated based on placing the CG of the section at the CG of the segment. This file is saved on a scratch disk file so subsequent manipulations may be made without having to rescale the data. Before drawing the image, a dummy display item is created out of viewing range. This is because subroutine ROTMIR, used to display the openings, expects to replace the current opening display with a new one at a different angle or mirror image factor. The program

CALL PLOTXY (IX,IY,IP)

Draws a vector from the current position to absolute position IX, IY. IP = 0 = pen up; IP = 1 = pen down.

Subroutine PLTSEG(N,K,IX)

For a die with N openings, PLTSEG creates an array, IX, of display items which the user may wish to hard copy plot using PLOTIT. The output array, IX, will contain K display item tags. The items put in the output array are the container circle, the center of the container, the spokes of the die, the CG of each segment, each opening, and the CG of each opening.

PLTSEG is part of module OVRL2D.

Subroutine PRPROS

PRPROS is used to read section data from a data file, calculate section parameters and output the results. As its first step, it asks if details are to be sent to the printer file. These details are various items calculated as part of the operation of ALEXTR such as the screen coordinates of a display item. Such values are not of interest to the general user, but rather are used by a programmer to track the operation of the system. Thus, the usual answer to the query is No or the default CR.

READIN is then called to get the section coordinate data from the disk. If any problems are encountered, the error flag is set and a return made to ALEXTR. If there are no errors, INTRPL is called to interpolate the section polygon, XSECA is called to calculate section properties, such as area, perimeter and center of gravity, and CRCMCL is used to find the circumscribing circle. The results of these routines are output to the CRT and print file by PARAMS.

This program element is contained in module OVRL1A.

does not know whether or not the latest display item is an opening. It simply deletes the last item and replaces it with an opening display at the proper position and orientation. The mirror image flag is set to -1. When ROTMIR is called, this flag will be negated to yield +1 with the result that a mirror image operation is not made. After the opening is displayed in the segment, TESTIT is called to examine the tolerance between the opening and the edge of the segment.

PLACEM is part of module OVRL1F.

Subroutine PLOTIT (N,M)

PLOTIT determines if a hard copy plot of the CRT display is desired. If any response other than Y(yes) is given by the user, no plot will be made. If a plot is desired, the N items contained in array M are copied onto the X,Y recorder. In addition, a unit vector representing 1.0 inch to the appropriate scale of the display is drawn at the bottom of the page.

This subroutine is contained in module OVRL2A.

Module PLOTOB

PLOTOB contains Macro-coded FORTRAN-callable subroutines used to drive the plotter. The subroutines included in PLOTOB are:

CALL COPYGT (ITAG,SCALE)

Copies the item specified by ITAG from the display file to the XY plotter.

CALL INITXY (IFLG,IX,IY)

Initializes the XY plotter and places the pen at IX,IY. This must be the first call to the XY plotter. IFLG should be initialized to Ø before the first call to INITXY. When INITXY is part of an overlay, IFLG should be defined in the root segment and carried in common.

Subroutine READIN

READIN is used to read section polygon coordinate data from a disk file. The file will be read until a match is found between the section number specified by the user (obtained by SETUPS) and a section number in the file. Each section has the following data format:

- First card (or line): Section number, I5 format, Section title or description, 75A1 format.
- Next N Cards: X, Y, R coordinate data; one card for each point on the polygon - 3F10.4 format
- Last Card: 9 in Columns 1-10.

In addition to the above data for each section, the very first line of the data file is used for text to describe the file. This is not used by ALEXTR but may be of value to a user when editing a data file.

READIN first initializes a counter, NP, of the number of points read to 1. The section number is read, followed by a read of the X, Y, R data. If the value of X is greater than 999, the end of section has been found. If X is less than 999, the value of NP is compared to 49. If

- NP < 49: error condition
- NP > 49: increment NP and read next card.

If more than the 49 allowable polygon coordinate points are read before the end of section (EOS) card is found, the error code is set. However, the file continues to be read until the end of section is found in order to prepare for the next section. When the EOS or end of file (EOF) is found, the program returns.

When the EOS is found with fewer than 50 points, NP is decremented as the last value read did not represent data, but was the EOS marker itself. If NP is less than 3, an error flag is set as at least 3 points are needed to describe a section. If $3 \leq NP \leq 49$, the value is correct. The section number

Subroutine PRSCHR

PRSCHR obtains predefined values for the characteristics of an extrusion press. The user is requested to enter the identification number of the press. At present, three presses have been defined so a valid user response is 1, 2 or 3. PRSCHR then checks that at least a one-hole die can be used on this press using the criterion that the container diameter must be at least one inch larger than the circumscribing circle diameter. If this condition is not satisfied, the error flag is set and the subroutine terminated.

If the press meets this criterion, the following characteristics for the press are obtained from a set of tables:

(1) Press capacity (tons)
(2) Maximum billet length (inches)
(3) Normal butt length (inches)
(4) Maximum number of die openings
(5) Runout table length (feet)
(6) Cycle time (seconds). This is the time from when one extrusion stops until the next one starts and would include removing the butt, cleaning the die, and charging and crushing a new billet.

The material properties of flow stress and extrusion speed are then obtained by a call to MATERL. The user is then asked to confirm, by using the default CR response, all press and alloy characteristics or to enter the desired values for each.

This is followed by calculating the billet section area, billet weight, and the single opening extrusion ratio. A report is then generated which summarizes the press characteristics to be used in subsequent computations. This report is sent to both the CRT and printer file.

It should be noted that the data defining press characteristics were made up by Battelle in order to test the programs. It is the user's responsibility to edit this data so that the characteristics of his presses are contained in the tables.

PRSCHR is part of module OVRL1D.

Subroutine REPRT2(NH1,NH2)

When calculating the number of die openings based on maximum yield, two values are calculated. The first is the number of openings based on the maximum yield from a billet of the specified diameter and length. The second is the number of openings which will give the maximum production rate in terms of pounds per hour. If these two values are not the same, REPRT2 is called. This reports the number of openings for each condition. The user is asked to select the value he wishes to use. If he selects neither value, he is asked to reconfirm the number he specified.

The parameters in the calling sequence are:

(1) NH1: The number of die openings based on maximum billet yield (input). The number of die openings selected by the user (output).

(2) NH2: The number of die openings based on maximum production rate.

REPRT2 is part of module OVRL2C.

Subroutine ROTMIR (I,J,IX,IY,N,K)

ROTMIR rotates or mirrors the display file of a die opening. This is used to create the display file of the opening at a new rotation angle or opposite mirror image.

When called, ROTMIR first determines if the data is to be mirrored or rotated. If rotation is specified, the user is asked to enter the incremental rotation angle. If Ø (or a carriage return) is entered, the points which define the opening polygon are illuminated as light pen sensitive items. GETHIT is then used to obtain hits on two points. The angle of the line between these two points is found. This is then used to determine the new absolute rotation of the opening which results when the line between the two points is set horizontal.

for the data just read is then compared to the desired section number. If the two are equal, the data is for the section desired. If the two section numbers are not the same, the entire process is repeated. When READIN terminates normally, it returns the polygon coordinate data as arrays X, Y and RR and the number of points as variable NP. All of these are returned via Common Block SECTN.

READIN is part of module OVRL1A.

Subroutine REPRT1
(N,JHOLES,LONGEX,ABLTLG,YIELD,ILONGS,MULTIG,EXRAT)

EPRT1 sends a report to the CRT and line printer. The title of the report is based on the value of N, as follows:

N	Title
1	Best Yield With Specified Openings
2	Best Yield With Maximum Length
3	Maximum Billet Yield
4	Maximum Pounds Per Hour

The other variables in the calling sequence represent the following reported values.

(1) JHOLES: Number of die openings
(2) LONGEX: The total extruded length of each piece
(3) ABLTLG: The billet length
(4) YIELD: The net percentage yield of usable extrusion from the billet, considering butt, breakthrough, and stretcher losses.
(5) EXRAT: The extrusion ratio
(6) ILONGS: The integer number of finished pieces will be produced from the billet
(7) MULTLG: The length of each finished piece.

REPRT1 is part of module CVRL2C.

operating conditions. The time to run one complete extrusion cycle is the sum of the extrusion time plus the time required to crop the butt, lubricate and clean the die, and load and crush a new billet. The inverse of the cycle time is the number of press cycles per hour. The gross billet weight per hour is the weight of a single billet times the number of cycles per hour. The net weight per hour is given by:

$$NW = A_e \ (N) \ (L_e - L_o) \ (C)$$

where A_e = area of one extrusion

N = number of die openings

L_e = total extruded length

L_o = breakthrough and stretcher length loss

C = number of cycles per hour.

If IFLAG is set to sero, the results of these computations are reported on the CRT and the printer file.

The parameters in the calling sequence are:

(1) NH: Number of openings in the die (input).

(2) LONGEX: The total length of the extrusion (input).

(3) BLTLG: The length of the billet to be used (input).

(4) IFLAG: A report/no report flag (input).
If IFLAG = 0, report is generated.
If IFLAG ≠ 0, no report is generated.

(5) WGTNET: The net weight of usable product extruded per hour (output).

RUNIT is part of module OVRL2C.

The angle, either computed or as entered by the user, is added to the current absolute rotation angle for the opening specified to get the new absolute rotated position. This angle is maintained in degrees in the range $\pm \pi$.

If a multi-hole display is being modified, ROTMIR returns to MODLYO at this point. If a single hole layout is being modified, the display file at the new rotation is found using SHFTIT, and the opening at the new position drawn.

If the mirror image was requested, the sign of the mirror image parameter is reversed. ROTMIR then proceeds as described above for a rotation operation, either returning if a multi-hole display is being modified, or generating the new display file for a single opening display.

The variables in the calling sequence are as follows:

(1) I: a flag. If I = 3, the mirror image is to be generated; if I $\neq$ 3, rotation is to be done (input).

(2) J: The opening to be modified (input).

(3) IX,IY: arrays of points describing the transformed display file for the opening (output).

(4) N: the number of coordinate points in the display file to be modified (output).

(5) K: a flag. If K = 1, a multi-hole display is being modified (input).

ROTMIR is part of module OVRL2E.

Subroutine RUNIT

(NH,LONGEX,BLTLG,IFLAG,WGTNET)

RUNIT calculates the number of extrusion cycles, billet pounds, and net extrusion pounds per hour which will be produced from a specified set of

Subroutine SETUPS (IJK)

When the value of IJK in the calling sequence is 1, SETUPS does the following:

(1) Initialize certain variables.

(2) Link the ALEXTR system to the graphics operating system software.

(3) Ask for the names and open or establish files for input, output, and print results.

(4) Read and write the input data file header.

(5) Request the operator to enter the number of the section to be processed. If the value entered is negative, the input data file is rewound and a dummy read made of the file header. This positions the file so that on the next read, the section number will be obtained.

IJK will only have a value of 1 when the program is initially started. When the value of IJK is 2, only step 5 above is executed.

When the value of IJK is 3, or IJK is 2 and NEWLYO is 1, the user is asked

"SAVE DATA FOR CAM NC (Y/N)".

If he responds Y(yes), he is asked to enter a file name. Data is saved in this file which can be used by program EXTCAM to generate the NC machining instructions for making the die.

IJK is 2 and NEWLYO is 1 when the user requests that data for a new section be read. IJK is 3 when the user requests that ALEXTR be terminated.

SETUPS is part of module OVRL1C.

Subroutine SHFTIT (JJ,IX,IY,N,JX,JY)

SHFTIT translates, mirrors and rotates a set of coordinates. The input coordinates are in CRT raster units relative to the user's origin.

Subroutine SCALZZ

(X,Y,IX,IY,J,K,XCTR,YCTR)

Subroutine SCALZZ converts elements J through K of arrays X and Y into arrays IX and IY. The input array represents points on the perimeter of some section as a series of closely spaced points. The input data would be in the user's dimension units, such as inches. The output data is in CRT raster units.

The scale factor used to scale the user units to raster units is determined elsewhere and is passed through common. The position of the scaled data is also translated so as to be relative to some point on the CRT. The coordinate values in raster units for this point are passed through common. The values for this point in the user's units are passed through the calling sequence variables.

After scaling and translating the points, a test is made to eliminate points which are unnesessarily close to each other. If a point is within three raster units in both X and Y of an adjacent point, the second point is eliminated. This reduces the amount of data which must be displayed without seriously degrading the quality of the image. The primary area where this elimination of points occurs is when a number of points are used to describe a radius.

The variables in the calling sequence are as follows:

- X,Y - input data arrays in user units.
- IX,IY - output data arrays in raster units.
- J,K - the first and last elements of arrays X,Y which are to be converted to IX,IY.
- XCTR,YCTR - the center, in user unit, about which the data is to be scaled.

This program element resides in the root segment, as part of module ALEXTR.

Subroutine TESTIT(TOLANC)

TESTIT generates two concentric circles superimposed on a cross. The outer circle diameter is scaled to the size specified by the user. The inner circle is half this size. The circles and cross are drawn as a single light pen sensitive display item. The major parts of this item are shown in Figure V-13, and are

(1) Invisible left vector, length = 1R
(2) Visible right vector, length = 2R
(3) Outer circle, 30 segments
(4) Invisible left vector, length = .5R
(5) Inner circle, 30 segments
(6) Invisible diagonal vector to bottom of outer circle
(7) Visible vertical vector to top of outer circle.

After generating this tolerance circle, the "END" light button is restored to view. The user is then free to move the tolerance circle by using the light pen wherever he wishes on the CRT. Since the entire tolerance circle is light pen sensitive, LGTPEN returns the coordinates of the circle where a LP hit is registered. MOVEIT then translates the center of the circle and therefore the entire item to the coordinates where the hit occurred.

The function of this tolerance circle is to allow the user to determine the relative distance between any two openings, or between an opening and any part of the die.

When "END" is hit with the LP the tolerance circle is deleted from the display file.

The parameter TOLANC is the default, or user-specified diameter of the outer circle in the user's dimensions.

TESTIT is part of module OVERL2E.

The position of the section CG is scaled into raster units and translated so as to be located relative to the center of the CRT.

The section origin is shifted so that the origin is at the section CG. The section coordinates are then rotated about the new origin. The mirror image is generated, if needed, by reversing the sign of the X coordinate of each pair. Finally, the section coordinates are translated so that they are positioned relative to the center of the CRT.

Subroutine STORIT (IX,IY,N)

STORIT creates a temporary file on disk and stores arrays IX and IY of length N in this file. It is called from PLACEM and is used to store the scaled display image file for an opening. Subsequent operations rewind and read this file and then operate on it as required to make the specified translation, rotation and mirror image changes.

STORIT is part of module OVRL1F.

Function TAN(X)

This function returns of tangent of the angle X. To avoid the possibility of division by zero, cosine(X) is determined first. Then

- If $\cos(X) = \emptyset$, $\mathrm{Tan}(X) = \pm 10^{30}$
- If $\cos(X) \neq 0$, $\mathrm{Tan}(X) = \sin(x)/\cos(X)$

For the condition of $\cos(X) = \emptyset$, the value assigned to TAN simply represents a very large number which lies in the range that the computer can handle. The sign of this number is the same as that of the input angle.

This program element resides in the root segment as part of module ALEXTR.

Subroutine TNGCMP (N,N2,TC)

TNGCMP is used to compensate tongues for their tendency to bend under load. When a tongue bends, it will tend to close the extrusion opening unless steps are taken to compensate for this effect. This tendency for an opening to close under load is shown in Figure V-8b.

Referring to Figure V-14, the end points, A and N of a tongue axis will have been indicated by the user in subroutine DISTRS. TNGCMP first determines the normalized coefficients of the tongue axis using LINEQ, and the angle of the axis, α. The angle of the normal, β, is found as:

$$\beta = \alpha + \pi/2 \quad . \tag{V-11}$$

All points between the axis end points are then shifted towards the axis along the normal to the axis. The normal distance from a point to a line is found as:

$$D = C1 * X + C2 * Y - C3, \tag{V-12}$$

where X and Y are the coordinates of the point, and C1, C2 and C3 are the coefficients of the line equation. A tongue point is then shifted along the normal by an amount equal to the product of its normal distance and the tongue compensation factor. The latter is expressed in inches per inch, and may be entered by the user if the default value is not chosen.

The results of this operation is shown in Figure V-14 where the points B and C, between the tongue axis points, are shifted to B´ and C´.

The variables in the calling sequence are:

(1) N1, N2: The indices to the points defining a tongue axis

(2) TC: The tongue deflection compensation factor.

TNGCMP is part of OVRL2H.

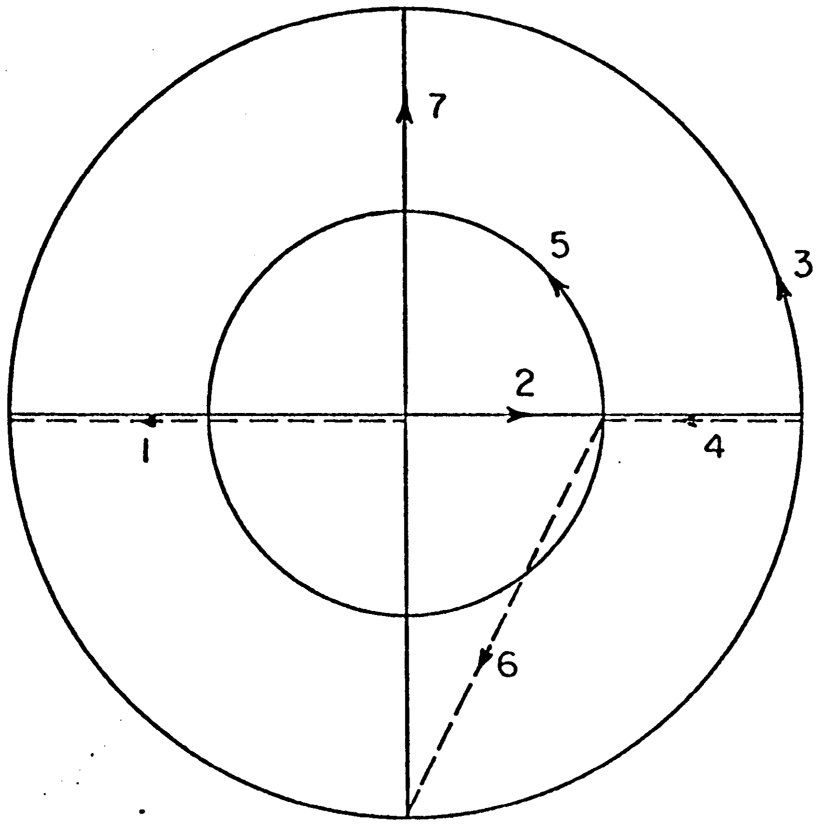


FIGURE V-13. VECTORS USED TO CREATE TOLERANCE CIRCLE
(Solid are visible lines, dashed are invisible)

Subroutine TNGSTR

(P,DT,BAT,BOT,CDB,CBB,TA,TW)

TNGSTR calculates the pressures acting on tool tongues and the resultant stress in each tongue due to bending and shear. A complete derivation of the equations used is given in Chapter 2, Volume 1. Within TNGSTR, the following functions are defined.

$$\mathrm{PLOAD}(I_1,I_2,L_1,L_2) = \frac{I_1}{I_2}\left[\frac{3L_2^{\,2}}{L_2^{\,4} + 6L_1^{\,2}\,L_2^{\,2} - 4L_1^{\,2}\,L_2^{\,2}}\right] \qquad \text{(V-13)}$$

where PLOAD = coefficient used to determine net pressure acting on tool #1

I_1 = moment of inertia of tool #1

I_2 = moment of inertia of tool #2

L_1 = length of tool #1 tongue

L_2 = length of tool #2 tongue

and

$$\mathrm{SIGSTR}\ (P,L,T) = [(3PL^2/T^2)^2 + (3PL/T)^2]^{1/2} \qquad \text{(V-14)}$$

where SIGSTR = total stress acting on a tongue due to both bending and shear

P = pressure acting on the tongue

L = length of tongue

T = thickness of tongue (tool thickness) .

After these functions are defined, TNGSTR determines the moment of inertia of each tongue. In the analysis, each tongue is treated as a rectangular

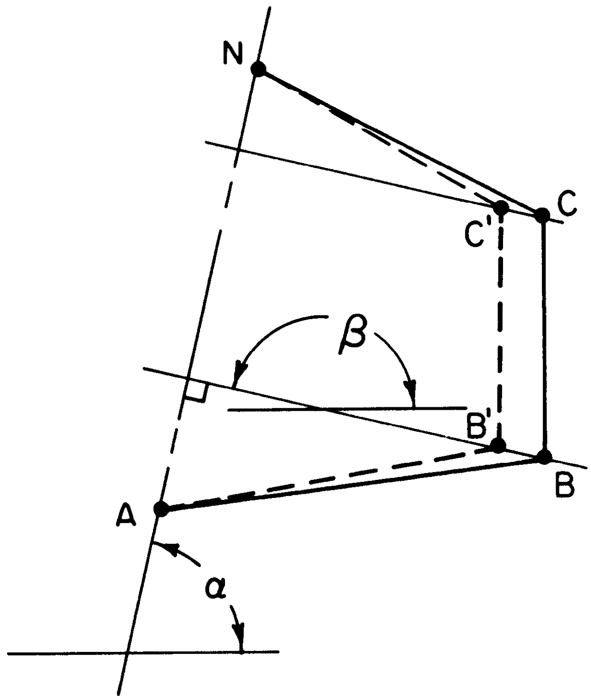


FIGURE V-14. TONGUE BENDING COMPENSATIONS

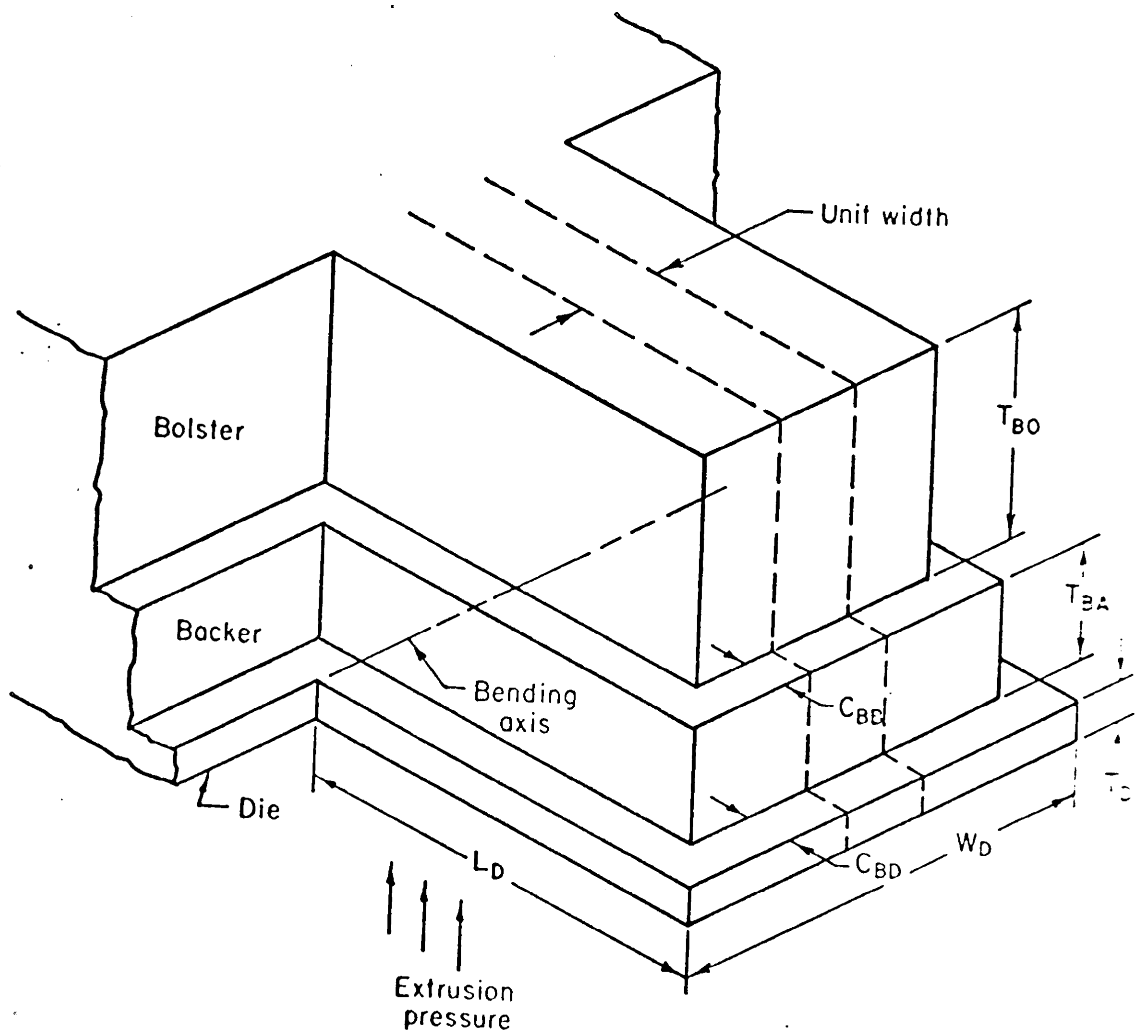


FIGURE V-15. TONGUE MODEL USED FOR STRESS ANALYSIS beam, cantilevered from one end. The moment of inertia of a rectangle is given by

$$I = BT^3/12. \qquad \text{(V-15)}$$

In the subsequent analysis, the pressures and resultant stresses are based on considering the tongues to be of unit width. This results in B in Equation (s) being set to 1, and then

$$I_i = T_i^3/12 \qquad \text{(V-16)}$$

where T_i = thickness of tool i.

Figure V-15 illustrates the tongue dimensions used for this analysis. The length of the die tongue, L_D, is found by

$$L_D = A_D/W_D \qquad \text{(V-17)}$$

where A_D = area of the die tongue

W_D = width of the die tongue.

Thus, regardless of its true shape, the die tongue is modeled as a rectangle of area A_D and width W_D, with the result that the rectangle has a length L_D. The length of the backer and bolster are found by subtracting the clearance or

$$L_{BA} = L_D - C_{DB}$$

$$L_{BO} = L_D - C_{DB} - C_{BB} = L_{BA} - C_{BB}$$

where L_{BA} = backer tongue equivalent length

L_{BO} = bolster tongue equivalent length

C_{DB} = clearance between die and backer

C_{BB} = clearance between backer and bolster.

1. Die thickness
2. Backer thickness
3. Bolster thickness
4. Clearance, die to backer
5. Clearance, backer to bolster.

The proper default value for the press previously indicated by the user are retrieved from the table and displayed, one at a time, to the user. The user may accept the default value or enter a new value. Limits of reasonability are tested for all five items. A value entered by the user cannot be less than one-fourth nor more than twice the default thickness of the respective tool. Limits on the clearances are not less than zero for either clearance; not more than one-inch for the die-backer clearances, and not more than two-inches for the backer-bolster clearance.

The arguments in the calling sequence are:

1. NP: The press identification number (input)
2. DT: Die thickness, inches (output)
3. BAT: Backer thickness, inches (output)
4. BOT: Bolster thickness, inches (output)
5. CDB: Die to backer clearance, inches (output)
6. CBB: Backer to bolster clearance, inches (output).

TOOLCH is part of module OVRL2G.

Subroutine TRNSIT(JJ)

TRNSIT is used to manually translate an opening on the die. It may be used in lieu of specifically stating the amount of translation to be made.

When first called, TRNSIT makes all items non-sensitive to light pen hits. It then calculates the display item tags for the opening to be moved and its CG "X" mark. The position of the latter item is determined and a tracking square is drawn at this position. This square is 30 raster units on a side and has a "+" and "X" superimposed on it. This tracking square is drawn as a single display item and is light pen sensitive.

L_{BA} and L_{BO} are tested for less than zero conditions. If either is less than zero, it is set equal to zero. The net pressure on each tool is then determined, considering the following conditions:

(1) Die, backer and bolster tongues exist
(2) Only die and backer tongues exist (i.e., non-conforming bolster)
(3) Die only exists.

The total stress due to (1) bending about the base of the tongue and (2) shear due to the net pressure acting on the tongue is found for each tongue, as appropriate. These values are reported and then the calculations are repeated for the cases of no bolster, and no backer or bolster.

The arguments in the calling sequence are:

(1) P: Average pressure across the die
(2) DT: Die thickness (inches)
(3) BAT: Backer thickness (inches)
(4) BOT: Bolster thickness (inches)
(5) CDB: Clearance, die to backer (inches)
(6) CBB: Clearance, backer to bolster (inches)
(7) TA: Area of tongue (inches^2)
(8) TW: Tongue width (inches)

TNGSTR is part of module OVERL2F.

Subroutine TOOLCH

(NP,DT,BAT,BOT,CDB,CBB)

TOOLCH obtains tool characteristics from a table and asks the user to accept these as default values or to enter new values. Each press which has defined characteristics in subroutine PRSCHR has tool characteristics in TOOLCH. The tool characteristics contained in the tables are:

The variables in the calling sequence are:

(1) XX,XY - coordinate points, before and after conversion (input and output)

(2) NB - the number of coordinate points to convert.

UNSCAL is part of module OVRL2D.

Subroutine XSECA

(X,Y,N,AREA,PERIM,CGX,CGY)

Given arrays X and Y, containing N points which represent points on the perimeter of a section, XSECA finds the area, perimeter and centroid for the section. The basis of the techniques used to find these values is given in Appendix II.

The variables in the calling sequence are as follows:

- X,Y: Arrays of input coordinate data representing the perimeter of a section.
- N : The number of coordinate data points.
- AREA: The area, in user units, of the section.
- PERIM: The perimeter of the section.
- CGX,CGY: The coordinates of the center of gravity of the section with respect to the X and Y axis, respectively.

XSECA is part of module OVRL1A.

Function XYINTR

(X1,Y1,X2,Y2,X3)

This function returns the second coordinate of a point which lies between the end points (X1, Y1) and X2, Y2), when the first coordinate of the point, X3, is known. The unknown coordinate is found by linearly interpolating between the end points. The equation is in the form

To move the opening, a light pen hit is requested on the tracking square. The tracking square is moved to the position where the hit was recorded. The current position of the CG "X" mark is found and the incremental move from the current position to the new position determined. The CG mark is then moved to the new position. The current position of the opening itself is then found, the incremental move added to it, followed by moving the opening to the new absolute position.

The above procedures are done in raster unit dimensions. The incremental move is also scaled to the user's dimensions, added to the previous absolute position and written to the CRT. The entire process of obtaining a LP hit on the tracking square, moving the square, the CG mark and the opening and reporting the current position in absolute user dimensions, operates in a continuous loop until the "END" light button text is picked.

The variable JJ in the calling sequence is the index to the opening to be moved.

TRNSIT is part of module OVRL2F.

Subroutine TRNSLT(I)

TRNSLT translates the display item of a die opening by the amount specified by the user. The user is asked to enter the amount of translation by entering two values in response to the query "DX,DY?"

The coordinates of the center of gravity for the opening are then incremented by the amount specified. This amount is converted to raster units by multiplying by the scale factor of the display. The current position of the opening is found by GTPSTN and then the entire display item is moved using MOVEIT and the scaled increment. The CG of the opening is found and moved by the same process. If the user enters Ø,Ø as the amount the opening is to be moved, subroutine TRNSLT is called. This allows the user to move the opening freehand by means of the light pen.

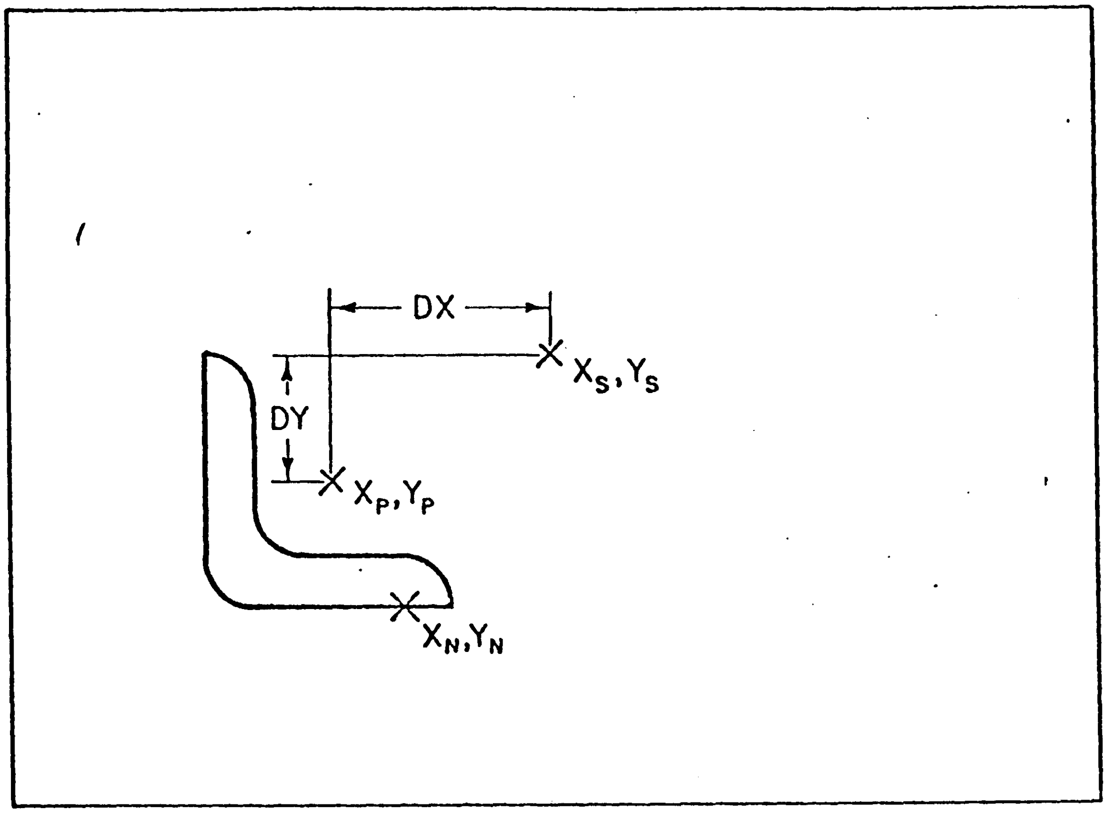


FIGURE V-16: CONVERSION OF CRT COORDINATES TO USER'S COORDINATE SYSTEM

$$Y3 = X1 + \frac{(X3 - X1)}{(X2 - X1)} * (Y2 - Y1). \quad (V\text{-}20)$$

To find the X-coordinate when the Y-co-ordinate is known, the calling sequence is

$$X3 = XYINTR\ (X1,Y1,X2,Y2,Y3). \quad (V\text{-}21)$$

To find the Y-coordinate when the X-coordinate is known, the calling sequence is

$$Y3 = XYINTR\ (Y1,X1,Y2,X2,X3)\ . \quad (V\text{-}22)$$

XYINTR is part of module OVRL2D.

$$K = .78/(.5(C-2)) \quad , \text{ if } C \geq 9 \qquad \text{(VI-1a)}$$

$$K = .78/(.4C) \quad , \text{ if } C < 9 \qquad \text{(VI-1b)}$$

where

K = bearing size constant

C = container diameter in inches.

The data in the scratch file is rewound and read to calculate the exact relative bearing length for each position of each die opening. The relative bearing length is found as:

$$Z_i = T\ (.62 + K * DR) \qquad \text{(VI-2}$$

where

Z_i = relative length of bearing at point X_i, Y_i.

K = die bearing constant.

T = Section thickness for calculating the bearings at this point.

DR = shortest distance from the current point to the container.

As the bearing at each point around an opening is found, it is tested to determine if it is a new minimum. The shortest bearing calculated for any opening is used to determine the absolute bearing lengths. The X, Y and Z arrays, with the exact relative bearing lengths for each position of each opening are saved on a second scratch data file. When the bearings for all openings and the minimum bearing of any opening are found, this second file is rewound and the data is read back in for each opening, one opening at a time. The user is requested to enter the cutter offset from the actual part contour, and the minimum bearing desired. The cutter size and cutter offset are used to correct the depth of cut. Assuming a ball-end cutter is used, the depth of cut of the ball will be:

$$D = (R^2 - W^2)^{1/2} \qquad \text{(VI-3)}$$

where

APPENDIX VI

DESCRIPTION OF COMPUTER PROGRAMS FOR EXTCAM

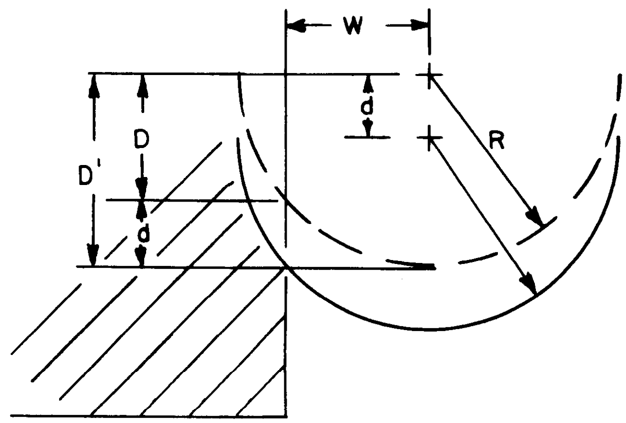


FIGURE VI-1. CORRECTION OF CUTTER DEPTH FOR CUTTER OFFSET IN BRNGCT

APPENDIX VI

DESCRIPTION OF COMPUTER PROGRAMS FOR EXTCAM

Subroutine BRNGCT

BRNGCT calaulates the tool path coordinates necessary to machine the bearings into the back of a die. The bearing, or die land, is the part of the die across which the extrusion slides as it is forced through the die. In order to compensate for different section sizes in the same opening, and the varying pressure distribution across the die, the bearings are of different lengths to provide uniform flow across the entire opening.

When called, BRNGCT first shifts the die polygon and bearing coordinates so that the origin is at the center of gravity of the opening. The first two die polygon points are copied as the last two so that the opening will close on itself when interpolated. INTRPL is then called to generate the discrete points which define the actual opening in the X-Y plane.

The user is next asked to enter the cutter size. The valid range is 0.0 to 1.0 inch. If the number of bearings defined for the section is more than 1, INSRTB is called. INSRTB inserts the bearing coordinates into their proper location in the interpolated die polygon. The relative thickness of the die at each interpolated point is generated as a Z-axis array. This is done by moving along the die, and assigning each point the thickness of the current bearing. When a match is found between a point on the die opening and a bearing position, the current bearing thickness is advanced to the next thickness in the list. The X, Y, Z arrays defining the die opening in general form are stored on a scratch disk file. This data will be subsequently retrieved and modified to define each opening exactly at its specified location and orientation on the die.

A constant for calculating the actual bearing size is found as follows:

first point on the part, and is then fed downward 1.0 inch. The cutter proceeds to move along the opening cutter path while at the same time feeding down into the work. This produces a ramp cut to the final depth and avoids generating a dwell mark. A dwell mark would be produced if the tool was fed to full depth at a single position. The ramp occurs over a distance of 1.0 inch in the X-Y plane. The depth of cut of the ramp is 1.0 inch for electrodes, and 55 percent of the die thickness when cutting the die openings directly.

As each opening or projection is completed, the tool is ramped back out of the work and then retracted to the Z = 0 plane. The tool is then moved to the first position of the next opening and the process repeated. When the last opening is completed, the tool is returned to the center of the die or electrode blank. It should be noted that when electrodes are cut, the extra material between the electrode projections is not machined out. This would be done manually after the projections were defined by NC machining.

The parameter JJ in the calling sequence indicates whether a die opening or electrode projection is to be cut. If JJ = 2, the die openings are cut; if JJ $\neq$ 2, the electrode projections are machined.

DIECUT runs at the first overlay level.

PROGRAM EXTCAM

EXTCAM is the root segment of the EXTCAM extrusion die manufacturing system. It does little data processing of its own. Instead, it handles the reading of the data file created in ALEXTR and then calls the appropriate subroutine depending on the operator's indication of what is to be done.

When first started, EXTCAM asks the user to enter the data file name to be used. This file, an output from ALEXTR, is then opened and read. The user is next asked if the die dimensions are to be expanded due to thermal and stretch considerations. The expansion coefficient is calculated and used as indicated by the user.

The user is then asked what machining operation tool path is to be generated. The options are:

R = cutter radius

W = cutter offset.

In order to achieve a depth of cut D' at the bearing, the cutter must be lowered an additional distance, d, equal to

$$d = R - D. \tag{VI-4}$$

This situation is illustrated in Figure VI-1. The minimum bearing which was previously found and the adjustment for cutter offset are combined in a single term. The fact that the bearing length is calculated from the front of the die, but the bearing is cut from the back is then considered along with the original height of the cutter above the back surface to arrive at a constant by which all bearings depths are corrected. This results in the actual depth to which the cutter is moved at each point along each opening. After determining the depth of each point, the X-Y coordinates are corrected for the cutter offset by using NEWPLY.

When the X, Y, Z coordinates for each opening are determined, they are sent to the NC tape image output file specified by the user. After all data have been sent to this file, it is closed, and the two scratch files are deleted.

Subroutine CENTER
(X,Z,R,XC,ZC,A,S,IFLAG)

CENTER in EXTCAM is the same as is used in ALEXTR with one exception. The variable IFLAG has been added to CENTER as used in EXTCAM. If IFLAG = 0, CENTER terminates after determining the sign of rotation of the angle at a corner or fillet. This sign is appended to the radius at the point, thus giving fillets positive radii and corners negative radii.

(11) The rotation, mirror image, and position of each opening in the die. The position data is the position of the centroid of the opening relative to the center of the die.

(12) The number of bearing positions specified.

(13) The position of each bearing and the die thickness at each point.

Subroutine FITARC

FITARC in EXTCAM is the same subroutine as is used in ALEXTR. It is described in detail in Appendix II. FITARC is part of module FITCTR.

Subroutine INITNC(TITLE,X,Y,Z)

INITNC is the first subroutine called when an NC tape output file is to be created. It first initializes the origin for subsequent incremental moves to the X,Y,Z position specified in the calling sequence. It then asks the user to input the name the user wants given to the output file, and this file is established in direct access, binary format. Its last operation is to send the first 72 characters (bytes) of array TITLE to the output file via a call to PARTNO.

INITNC is part of module EXTRNC.

Subroutine INTRPL

INTRPL in EXTCAM is the same subroutine as is used in ALEXTR. It is described in detail in Appendix II. INTRPL is part of module FITCTR.

Subroutine INSRTB (AX,AY,ND,DZ,CS)

INSRTB is used to add two new coordinates to an interpolated array. These new points signify where a depth change is to occur because a new bearing

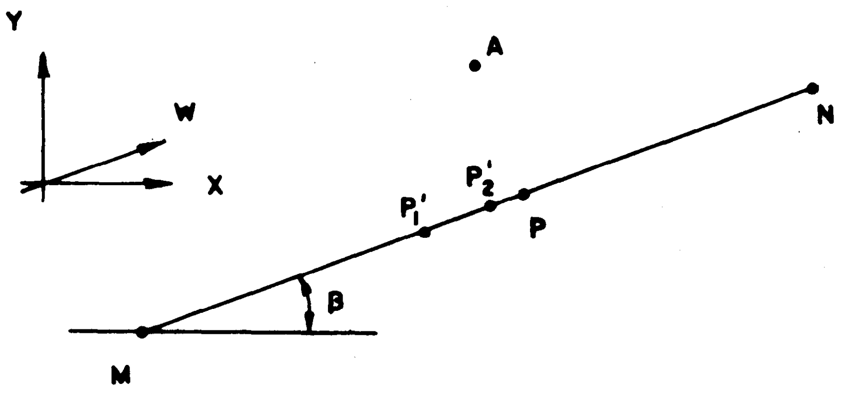
Subroutine DIECUT(JJ)

DIECUT can be used to machine the openings directly into the front or back of a die, or to machine the electrode for subsequent EDM'ing the openings from the front. The operation of DIECUT is quite similar to that of TEMPLT. However, DIECUT operates on the die polygon data, corrected for thermal and stretch effects as specified by the user, while TEMPLT works with the original extrusion polygon data.

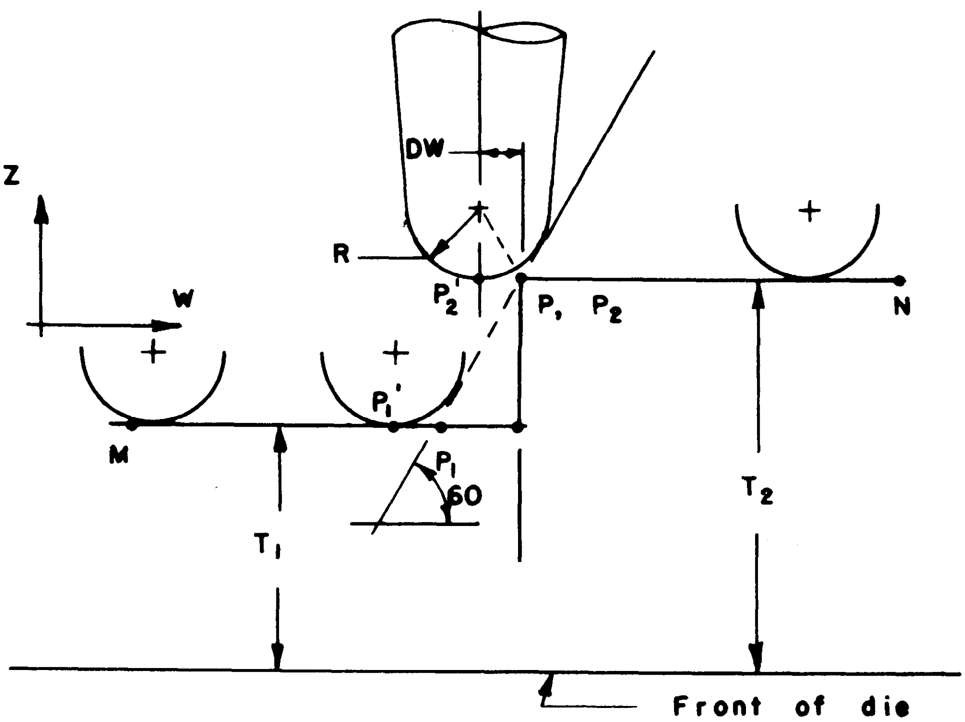
When called, DIECUT first requests the cutter size from the user. If an EDM electrode is to be cut, the burn and polish allowance is also requested. The cutter radius is reduced by this allowance, with the result being that the electrode is cut slightly smaller than the actual die opening. When the electrode is used to EDM the openings, the undersize electrode will give an on-size opening. If the die opening is to be machined, the cutter size is set negative to cut inside the die polygon; if the electrode is to be cut, the cutter size is positive. The proper tool path polygon is determined in NEWPLY, using the signed cutter size passed to NEWPLY in the calling sequence. After NEWPLY, the first six polygon points are copied as the last six so that the cutter will go around the part more than once. The actual cutter position coordinates are then found using INTRPL. The interpolated data is saved on a scratch file since it will be rotated, translated, and mirrored as necessary for each opening of the die or electrode.

If a die is to be machined, the user is asked if the die is cut from the front or the back. If the die is cut from the back, a flag is set to -1 in order to generate the mirror image of the front opening. When electrodes are cut, this same flag is always set negative. This is because the electrode is cut upside down, and then inverted in the actual EDM process.

The next operation is to determine the actual cutter locations for each opening in the die, or each projection of the electrode. In order to ensure proper positioning of the electrode, the projections for a multi-hole die would be cut from a single piece of graphite and would remain joined at the base. Using the interpolated data previously calculated and saved on a scratch file, the cutter path coordinates for each opening are rotated, translated, and mirrored as necessary. The cutter is moved in the X-Y plane to the



a) In X-Y plane



b) In W-Z plane

FIGURE VI-2. ADDING BEARING TRANSITION POINTS TO INTERPOLATED DIE OPENING POLYGON

(1) Extrusion template. This is a template of the finished extrusion, as specified by the customer.

(2) Die. This will machine the extrusion die profile from the front or back of the die. The profile is machined at a constant depth. This operation could be used as a roughing cut before machining the bearings or EDM'ing the finished die opening.

(3) Electrode. This will machine the EDM electrode needed to cut the die opening. The EDM operation is assumed to always occur from the front of the die.

(4) Bearings. This calculates the cutter path necessary to produce the bearings (die land) in the back of the die.

After a cutter path is generated as requested by the user, EXTCAM loops back and re-reads the input file. The user may then specify a different machining operation or cutter size. To produce cutter paths for a different die, EXTCAM must be terminated and then re-run.

The data file containing the extrusion and die coordinated in unformatted binary format. As such, it cannot be read by the EDITOR and appear in user-readable form. The data file used by EXTCAM contains the following information:

(1) Die title -- 72 characters.

(2) Centroid of extrusion relative to user's original origin.

(3) Die thickness (inches).

(4) Coefficient of thermal expansion and typical extrusion temperature.

(5) Container diameter.

(6) The number of points defining the finished extrusion.

(7) The X, Y and R arrays defining the finished extrusion.

(8) The number of points defining the die.

(9) The X, Y and R arrays defining the die.

(10) The number of openings in the die.

disk file. When NCOUT is called, the character being output (IC) is stored in the buffer. When the buffer is full (512 characters), the buffer is copied to the output file and then cleared.

NCOUT is part of module EXTRNC.

Subroutine NEWPLY (XI,YI,RL,XO,YO,RO,C)

NEWPLY finds the cutter path coordinates which will generate a part in the X-Y plane when using a cutter of radius C. Before finding the offset cutter path, two tests are made. The first is a check for a cutter of Ø.Ø size. This could be specified when the user wanted to scribe or plot the actual part outline. If the cutter size is Ø.Ø, the offset points are identical to the original input points.

To find the offset at a point, three points must be used. These are the point in question itself plus the points immediately adjacent on either side. If these three points lie in a straight-line, the offset at the second point is normal to the line at a distance equal to the cutter radius.

If the two lines defined by the three consecutive points are not parallel, the offset will lie along the bisector of the angle formed between the two lines. Referring to Figure VI-3a, the angles of the two lines, relative to a horizontal, are given by

$$\theta_B = \tan^{-1} ((Z(1) - Z(2))/(Y(1) - Y(2)) \qquad \text{(VI-8)}$$

$$\theta_E = \tan^{-1} ((Z(3) - Z(2))/(Y(3) - Y(2)) \quad . \qquad \text{(VI-9)}$$

It should be noted that the values returned for θ_B and θ_E have direction as well as magnitude. The angle of the bisector of the included angle is

$$\theta_A = (\theta_B + \theta_E)/2 \quad . \qquad \text{(VI-10)}$$

Half of the angle subtended by the arc can be found as follows:

thickness has been specified. INSRTB is an extension of subroutine ADDPNT used in ALEXTR. INSRTB first finds the location of a point on the interpolated extrusion die perimeter which is closest to the bearing transition coordinate point AX, AY. Referring to Figure VI-2a, let P be the point on the perimeter which is closest to AX, AY in X, Y space.

Looking at line MPN from a point normal to the line, the view in W, Z space would appear as shown in Figure VI-2b. The thickness at M is T_i, and at N is T_2. The transition from T_1 to T_2 occurs at P. It was arbitrarily decided that the transition from T_1 to T_2 was to be completed by the time the cutter reached P. Furthermore, to avoid a step change in the bearing, the transition would be made at a ramp of 60°. If a step up ($T_2 > T_1$) is being generated, P would be replaced by P_2, and a new point P_1, ahead of P_2 would be generated. The position of P_1 in the W-Z plane would be:

$$P_1 = P_2 - (T_2 - T_1)/\tan(60). \qquad \text{(VI-5)}$$

Since the profile M,P_1,P_2,N is to be machined with a cutter of finite tip radius, further correction of P_1 and P_2 must be made. Along the W axis, P_1 and P_2 must be shifted by DW, where

$$DW = R\cos(60). \qquad \text{(VI-6)}$$

Thus,

$$P_1' = P_1 - DW \qquad \text{(VI-7a)}$$

and $$P_2' = P_2 - DW. \qquad \text{(VI-7b)}$$

The position of the two new points in X, Y space is then determined from the angle β of the original line in the X, Y plane. The Z coordinate thickness at P_1' is set to T_1, and to T_2 for P_2'. The thickness is transformed to the actual cutter depth as described in BRNGCT.

If a step down is involved ($T_2 < T_1$), the same general procedure is followed. In this case, however, P_1 is set to P, and P_2 is offset towards N.

$$\gamma = \pi/2 - (\theta_A - \theta_E) = \pi/2 - (\theta_B + \theta_E)/2 + \theta_E$$
$$\gamma = (\pi + \theta_E - \theta_E - \theta_B)/2 \quad . \tag{VI-11}$$

The offset values then become:

$$XO = X(2) + \frac{C}{\cos|\alpha|} \sin\theta_A \cdot D$$
$$YO = Y(2) + \frac{C}{\cos|\gamma|} \cos\theta_A \cdot D \quad , \tag{VI-12}$$

where D = +1 if γ is positive, D = -1 if γ is negative, and C is the cutter radius.

NEWPLY can be used to find the offset to a polygon either before or after the polygon is interpolated. When the offset is found before interpolation, it is also necessary to correct the radius at each fillet or corner by the size of the cutter radius. When a polygon is offset to the outside, all corners are increased by the cutter radius and all fillets are decreased by this amount. This is illustrated in Figure VI-36 where the radius R_2 of the fillet is decreased by C to R_2', and R_3 is increased by C to R_3'.

The variables in the calling sequence are:

(1) XI, YI - 3 element arrays defining the input coordinates (input).

(2) RI - The radius of the second input point (input).

(3) XO, YO - The offset coordinates corresponding to the second input point (output).

(4) RO - The adjusted radius for the offset point (output).

(5) C - The cutter radius for which the offset is to be found (input).

That is, in Equation VI-5, P_1 is known and P_2 is solved for. P_1' and P_2' are then found by reversing the sign in Equation (VI-7a) and (VI-7b).

The variables in the calling sequence are as follows:

(1) AX, AY: The coordinates of the point where the bearing transition is to take place (input); the first of the two points added to indicate where a transition occurred.

(2) ND: The number of coordinates in the interpolated arrays. ND (output = ND (input) + 2.

(3) DT: The difference in section thickness at the bearing transition point. If DZ < 1, step is down; if DZ > 1, step is up.

(4) CS: The cutter radius.

Subroutine INSRTB is contained in module INSRTB.

Subroutine LEADER(N)

LEADER uses NCOUT to output leader code to the output file. N frames of leader code are generated. LEADER is called at the start of PRTSRF, before and after the title is punched, at the completion of each Y-Z machining pass, and at the end of PRTSRF just before the NC file is closed. The ASCII code for leader is equivalent to a line feed (12_8).

Subroutine LINEQ

LINEQ in EXTCAM is the same subroutine as is used in ALEXTR. It is described in detail in Appendix V. LINEQ is part of module INSRTB.

Subroutine NCOUT(IC)

The subroutine stores all characters which are to be output to the NC tape file in a buffer. When the buffer is full, the buffer is copied to a on the tape, the machine tool will pause, until the "pause" button is pressed by the operator.

PAUSNC is part of module EXTRNC.

Subroutine PNCHNC(X,Y,Z,M)

PNCHNC formats absolute coordinate data into the incremental form required for the BCL CNC milling machine. To do this, it first converts the absolute value to an integer representing the size of the move in thousandths of an inch. That is, for an X-axis move

$$IX = \text{integer part } (1000 * X). \qquad \text{(VI-14)}$$

This integer, absolute value, is then converted to integer incremental form by subtracting the current value from the previous value as follows:

$$IDX = IX_{new} - IX_{old} \quad . \qquad \text{(VI-15)}$$

The old integer value is then replaced by the new integer value, and subroutine PXYZ is called to output the current move. When m = 1, the initial "old" values for the three axis are set, but no data is output.

After each set of X, Y, Z coordinates is processed, and end-of-block mark (i.e., carriage return, line feed) is appended to the output file using NCOUT.

To summarize the operation of the NC formatting routines, assume the following three(3) sets of absolute coordinates exist:

X	Y	Z
1.154	2.837	0.549
1.289	2.956	0.650
1.508	3.254	1.064

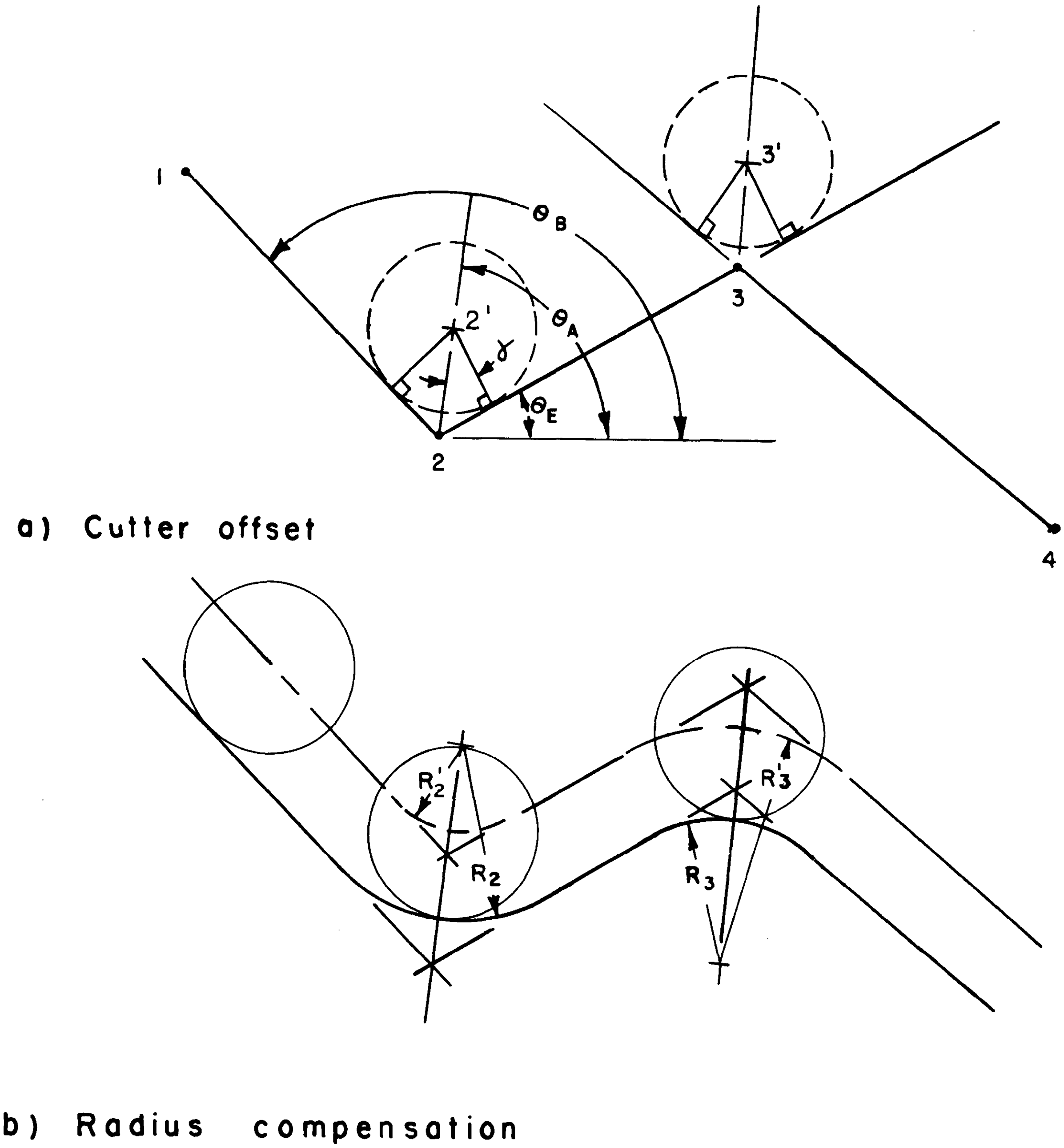


FIGURE VI-3. OFFSETTING COORDINATES TO CORRECT FOR CUTTER SIZE

$$INC_{New} = INC_{Old} - INT * 10000$$
$$= 1234 - 0 * 10000 \qquad (VI-17)$$
$$= 1234 \quad .$$

On the next pass, the power of 10 is reduced to 10^3 or 1000. Then,

$$INT = integer\ (1234/1000) = 1 \quad . \qquad (VI-18)$$

The ASCII representation of 1 (61_8) is sent to the output file and the new value becomes:

$$INC = 1234 - 1 * 1000$$
$$= 234 \quad . \qquad (VI-19)$$

his process repeats until all digits have been processed.

The variables in the calling sequence are as follows:

INC: The integer value for the distance to be moved along the axis NAME

NAME: The ASCII code for the axis along which a move of INC size is to be made.

PXYZ is part of module EXTRNC.

Subroutine RWNDNC

RWNDN codes an "M3Ø" onto the NC tape output file. This is interpreted by the CNC machine control as a command to rewind the NC tape.

RWNDNC is part of module RWNDNC.

Subroutine OFFSET(X,Y,XN,YN,CB,SB,D)

Given a point in X, Y space, OFFSET finds a second point which is a distance D from X,Y and whose sine and cosine, relative to X,Y are SB and CB, respectively.

That is:

$$XN = X + D * CB \qquad \text{(VI-13a)}$$

$$YN = Y + D * SB. \qquad \text{(VI-13b)}$$

OFFSET is part of module INSRTB.

Subroutine PARTNO(NCHAR,NPART)

PARTNO formats an ASCII text string for the output file. This text string could be the part name or description, or instructions to the operator. PARTNO first generates an "M80" code which indicates to the NC controller that what follows is text and not vector data. The characters in the text string are then output one at a time by using a Do-loop from 1 to NCHAR. An end-of-block code is output following the text.

The parameters in the calling sequence are as follows:

NCHAR: The number of characters in the text string which are to be output

NPART: A text string (byte array) which is to be output.

PARTNO is part of module EXTRNC.

Subroutine PAUSNC

PAUSNC produces an "MØØ" code on the NC tape output file. This is interpreted as a "pause" by the BCL/CNC control. When this code is encountered and simultaneously downward. The Z motion is at the rate of 0.4 inch/inch of forward travel in the X-Y plane. This dual forward and down motion is continued for an X-Y distance of 0.5 inch, or a Z-axis travel of 0.2 inch. When at full depth, the tool will be 1.150 inches below its original net point. If the tool is originally positioned 1.0 inch above a piece of 1/8-inch template stock, the depth of cut programmed will put the tool 0.025 inch below the bottom of the stock at full depth.

Once at full depth, the Z-position is repeated for all remaining X-Y cutter positions. A lead-out ramp is also generated starting from the last position and working backwards through the list. The lead-out ramp causes the tool to rise 0.2 inch over an X-Y distance of 0.5 inch. The cutter is then raised to the Ø.Ø depth plane, followed by a return to the origin. After outputting all position commands, the tape rewind command is sent to the tape file and then the file is closed. It should be noted that the NC tape is not physically generated in TEMPLT. Instead, a tape image file is created on the disk. This file may be read by NCDATA to graphically verify the data. The actual NC tape may be generated by copying the disk file onto paper tape using DEC's PIP utility program.

Subroutine TESTIT(J)

Given four consecutive points, starting with J, TESTIT checks to ensure that the second and third points lie between the first and fourth. The four points lie on a straight-line. Therefore,

$$L_{14} = L_{1N} + L_{N4} \qquad \text{(VI-21)}$$

where L is the length between the points referred to by the subscripts, and N refers to the second or third point. If this equality is not met within a small tolerance, the point being evaluated is assumed to have been computed incorrectly.

TESTIT is part of module INSTRTB.

The two(2) vectors represented by these values would appear in the output file as:

X135	Y119	Z101	(first vector)
X219	Y298	Z414	(second vector)

It should be noted that PNCHNC contains data tables for preparing tape in either EIA or ASCII format. By "commenting" the one table not desired, either format can be produced.

The variables in the calling sequence are as follows:

X,Y,Z: Arrays of absolute coordinates in user units

M: The number of elements in each of the above arrays.

PNCHNC is part of module EXTRNC.

Subroutine PXYZ(INC,NAME)

XYZ takes the integer data representing incremental motion commands for any of the three axes, converts the data to individual ASCIII or EIA characters, and then uses NCOUT to output the characters. The routine operates by repeatedly dividing the input value, INC, by successively smaller powers of 10. The quotient, if not a leading zero, is output via NCOUT. The quotient is next multiplied by the same power of 10 and subtracted from the initial value to get the next value.

For example, let INC = 1234.

Since the largest integer value which can be expressed is ±32768, the first power of 10 is 10^4 or 10000. Thus, the first digit to be output is

$$\text{INT} = \text{integer } (1234/100000 = 0 \quad . \qquad \text{(VI-16)}$$

Subroutine TEMPLT

TEMPLT generates the cutter path used to machine a template model of the finished extrusion. This operates on the extrusion data, not the die data, and no correction is made for thermal effects. TEMPLT starts by calling MAXMIN to determine the size of the rectangular blank from which the template will be cut. The blank size and the starting position of the cutter relative to the lower left corner are reported to the CRT and printer. The user is then asked to enter the cutter size to be used to cut the template. The value entered must be equal to, or lie between 0.0 and 1.0 inch.

TEMPLT then determines if the cutter specified is small enough to cut all the fillets. In EXTCAM, before TEMPLT was called, the sign of each radius was determined. Corners are considered negative and fillets are positive. When cutting a template, any size cutter can be used to cut any size corner. For fillets, however, the cutter radius must be less than or equal to the fillet radius, so that the fillet is fully developed. All fillets are tested, and an error message generated if any fillet is too small to be cut with the cutter size specified. If an error situation is encountered, the user is requested to enter another cutter size.

When the cutter size test is passed, NEWPLY is called to generate the cutter path polygon. The cutter path polygon is identical in shape to the part polygon. The difference between the two is that the cutter path polygon is uniformly larger on all sides of the part polygon by the size of the cutter radius. After NEWPLY is completed, the first six points of the cutter path polygon are duplicated as the last six. This makes the cutter path go around the part more than once, and provides distances over which the cutter can be ramped into and out of the work. INTRPL is then called to interpolate the polygon. This developes the individual points around the corners and fillets needed to define the radii as a series of straight lines.

After interpolation is completed, the actual tool positions are calculated, including the Z-axis (depth) commands. From the starting position over the center of gravity, the tool is moved in the X-Y plane to the first point on the interpolated cutter path. The tool is then given a down-feed command of 0.950 inch. The tool is next moved forward along the cutter path

Subroutine THERML(C,T)

THERML is used to enlarge the die to compensate for 1) thermal expansion of the extrusion during extrusion, and 2) the reduction in the extrusion size when being stretched for straightness after being extruded. The analysis of the thermal expansion of the die and extrusion is detailed in Appendix I. In THERML, an expansion coefficient is calculated and applied linearly to the X and Y dimensions of the die. This is used to account for the fact that most extrusions are made hot and will shrink when cooled to room temperature. In order to be at size when cool, the extrusion must be made slightly oversize when hot.

When first called, variable EXC in Common /DATUMS/ is zero. In this case, the user will be asked if he wishes to compensate the die for thermal and stretch effects. If the response is other than Yes, EXC is set to 1.0, and THERML is terminated. If the response is Yes, the expansion factor is determined using the coefficient of expansion and average extrusion temperature. These two items are properties of the alloy being extruded and would be obtained by a table look-up during processing by ALEXTR. The expansion factor is increased by 0.004 to allow for the shrink during stretch. The combined factor is then reported to the user who may accept it as the default value, or over-ride it with a value of his own. The combined factor generally is in the range of 1 to 2 percent.

Once the factor is determined, the X and Y die polygon and bearing position coordinates are expanded by the factor. This factor, EXC, will be automatically applied on subsequent passes through EXTCAM to the die and bearing dimensions. That is, after the first time THERML is called, it will not be called if EXC = 1.0 (no expansion correction is to be made). If EXC is greater than 1.0, the expansion will be made using the same factor as previously calculated or specified.

The variables in the calling sequence are:

(1) C: The coefficient of thermal expansion of the alloy to be extruded.

(2) T: The average temperature at which this alloy is usually extruded.

Labeled Common/NTRPTD/

(1) XI(300),YI(300): Arrays defining the perimeter of a section. This is developed by interpolating the input polygon data.

(2) NI: The number of points defining the perimeter.

Labeled Common/SECTN/

(1) X(51),Y(51),R(51): 3, 51 element arrays for the coordinates of the input polygon.

(2) NP: The number of coordinates defining the input polygon.

(3) NEXT: A logical flag used to indicate a new section is to be processed.

(4) GRPHCS: A logical flag used to indicate that the graphics system has been initialized.

(5) ISCTNO: The number of the section currently being searched for or processed.

(6) FS: The flow stress of the alloy indicated.

(7) AREA: The cross-sectional area of the extrusion.

(8) PERIM: The cross-sectional perimeter of the extrusion.

(9) CCX,CCY: The coordinates of the centroid of the section in the user's units.

(10) SHPFTR: The shape factor of the section (perimeter divided by weight per foot)

(11) CCD: The diameter of the smallest circle which will circumscribe the section.

(12) XCTR,YCTR: The center of the circumscribing circle.

(13) IERROR: A flag indicating an error in logic or operation has been found (=1, no error).

(14) NOPRES: The press system number specified by the user.

(15) PCAPAC: The press capacity (tons).

(16) BLTDIA: The extrusion billet diameter.

(17) BLTLGH: The extrusion billet length.

(18) BUTLGH: The length of the butt discarded at the end of the extrusion.

(19) NHOLES: The maximum number of openings permitted in a die for a particular press.

(20) LGRUNO: The runout table length for a press.

(21) BLTARE: The cross-sectional area of the billet.

(22) BRGLGH: The average bearing length.

(23) LONGPL: The finished length of a cut-to-length extrusion.

APPENDIX VII

VARIABLES ASSOCIATED WITH "COMMON" BLOCKS USED BY FORTRAN

Labeled Common/SPARE1/

(1) YES: A byte variable assigned the ASCII equivalent of Y. Used to determine the user's response to Yes/No queries.

(2) ANS: A byte variable containing the user's response to Yes/No software queries.

(3) IXYFLG: A flag which allows subroutine INITXY to be initialized.

Labeled Common/SYSPR/

(1) LUNKB: The keyboard logical unit number.

(2) LUNCRT: The CRT and/or typer logical unit number.

(3) LUNPR: The printer disk file logical unit number.

(4) LUNCR: The logical unit number for the input data file.

(5) LUNNC: The logical unit number for NC output data.

(6) PI: π. Numerical value 3.141592

(7) TWOPI: 2π. Numeric value 6.283184

(8) PIOVR2: $\pi/2$. Numeric value 1.570796

(9) IPRINT: A flag used to control printing of intermediate details of various computations. If IPRINT = 2, details are printed.

(10) ZERO: Numeric value Ø.Ø

(11) COEFEX: The coefficient of thermal expansion for the alloy specified.

(12) EXTEMP: The temperature at which the alloy specified is normally extruded.

"COMMON" USED BY EXTCAM

Labeled Common/CNSTNT/

(1) LUNKB: The keyboard logical unit number.

(2) LUNIN: The input data file logical unit number.

(3) LUNOUT: The NC tape file logical unit number.

(4) LUNPR: The printer logical unit number.

(5) PI: π. The numeric value 3.141.59.

(6) PIOVR2: $\pi/2$. The numeric value 1.57079.

APPENDIX VII

VARIABLES ASSOCIATED WITH "COMMON" BLOCKS USED BY FORTRAN

The following describes the variables associated with each "COMMON" block. "COMMON" blocks are used by FORTRAN as a means of communication between program elements, such as between subroutines at the same or different overlay levels.

"COMMON" USED BY ALEXTR

Labeled Common/BERNGS/

(1) XB(40), YB(40): Coordinates on the perimeter of a die where a bearing dimension is specified.

(2) BRNGSZ(40): An array defining the section thickness at a point on the perimeter of a die.

(3) NB: The number of bearings defined for a die.

(4) TITLE(20): An array used to store the section description or title.

Labeled Common/DISPLA/

(1) IXCTR,IYCTR: The origin of a cartesian coordinate system on the CRT. This value is in raster units and provides the relative starting location for generating a display.

(2) NLINES: Unused.

(3) LINE1: The vertical coordinate of the text scroller area at the bottom of the CRT display. This is set to 150 raster units in subroutine INITDS.

(4) SCALEF: The scale factor used to convert from the user's dimensions to CRT raster units.

(5) NTAG: The tag value of the next item to be displayed on the CRT.

(6) DISBFR: A 3000 word array used as the display buffer.

Named COMMON/PUNCHO/

(1) IXO,IYO,IZO: The origin to be used to determine all subsequent incremental position commands to the BCL/NC milling machine.

(2) NC: The count of characters in the NC tape output file.

(3) NR: The associated variable used in conjunction with the direct access format of the NC tape output file.

(4) IA: 512 byte array used to buffer the output to the NC tape file.

Named COMMON/SPACE/

(1) X,Y,Z: 500 element arrays used to store the cutter position coordinates before they are written to the NC tape output file.

(24) MULTLG: The integer number of pieces of length LONGPL which can be cut from the extrusion from a single opening.

(25) SPEED: The expected average extrusion speed for a specified alloy.

(26) LOSS: The length of extrusion from a single opening which is scrap due to breakthrough and stretcher effects.

(27) CBX,CBY: Unused.

(28) NOPENS: The number of openings in a die.

(29) I1,I2,I3: The indices to coordinates on the perimeter of the section which define the circumscribing circle. I1 and I2 define the longest of the 3 possible lines. If I3=0, the circumscribing circle is defined by the two points, I1 and I2.

(30) CONDIA: Extrusion press container diameter.

(31) NEWLYO: A flag, If ≠ Ø, an existing die layout is to be modified in subroutine PLACEM and MODLYO; if = Ø, a completely new layout is to be started.

Labeled Common/PLACES/

(1) THETA(10): The angular position of each opening of a die. This angle is the rotation of the opening about its center of gravity, relative to the original, input position of the opening.

(2) DXCG(10),DYCG(10): The position of the center of gravity of each opening relative to the geometric center of the die.

(3) MIRROR(10): A value indicating whether or not each opening of a die was mirror imaged.
If = 1, not mirror imaged.
If = -1, mirror imaged.

(4) XD(50),VD(50),RD(50): 3, 50 element arrays for the coordinates which define the die polygon.

(5) NDI: The number of coordinates which define the die polygon.

(7) TWOPI: 2π. The numeric value 3.28318.

(8) ZERO: Numeric value Ø.Ø.

(9) YES: A byte variable assigned the ASCIT equivalent of Y. Used to determine the user's response to Yes/No queries.

(10) ANS: A byte variable used to store the user's response to Yes/No queries.

Labeled Common/DATUMS/

(1) CGX,CGY: The coordinates of the center of gravity of an extrusion, relative to the user's coordinate system origin.

(2) DIETHK: The thickness of the die blank (inches).

(3) TITLE: A 72 character array describing the die (name, number, customer, etc.)

(4) NOPENS: The number of openings to be made in the die.

(5) THETA: An array giving the angle of rotation (in degrees) of each opening about its centroid.

(6) DXCG,DYCG: Arrays defining the location of the centroid of each opening relative to the center of the die.

(7) MIRROR: An array for the mirror image parameter for each opening.
If MIRROR = 1, opening is not mirrored
If MIRROR = - 1, opening is mirrored.

(8) XD,YD,RD: Arrays defining each point on the die polygon, and the radius at each point.

(9) NDI: The number of points defining the die polygon.

(10) XE,YE,RE: Arrays defining each point on the extrusion polygon, and the radius at each point.

(11) NEX: The number of points defining the extrusion polygon.

(12) XB,YB: Arrays defining the position on the die perimeter where a bearing length transition is to occur.

(13) BRNGSZ: An array defining the section thickness to be used for calculating the bearing length at each bearing transition point.

(14) NBR: The number of bearing transition points.

(15) EXC: The expansion coefficient applied to the die dimensions to compensate for thermal and stretch effects.

(16) CONDIA: The diameter of the container to be used when making the extrusion.

Tag No.	Description
8	A circle concentric to the container circle but smaller by .75 inch on the radius.

When one opening of a die with multiple openings is displayed, such as when starting a die layout, the following tags are used. (Assume the die is to have N openings.

Tag No.	Description
6	The pie-shaped display of 1/N of a full circle.
7	The center of gravity of this segment of a circle.
8	The opening scaled to the segment.
9	An "X" to locate the center of gravity of the opening.

When all openings of a multi-hole die are displayed, the following tags are used:

Tag No.	Description
6	The spokes which partition the container into N segments of equal size.
7	Opening #1.
8	The center of gravity of opening #1.
9	Opening #2.
10	The center of gravity of opening #2.
.	.
.	.
.	.
5+2*N	Opening #N
6+2*N	The center of gravity of opening #N.
7+2*N	The center of gravity of segment #1.
8+2*N	The center of gravity of segment #2.

APPENDIX VIII

DISPLAY ITEM IDENTIFIERS

APPENDIX VIII

DISPLAY ITEM IDENTIFIERS

Display items are identified by tag numbers. The tag number is generally the first variable in a call to one of the graphics subroutines used to manipulate a display item. The following describes the tag numbers used for the various display items in the ALEXTR system. These display items are global in that they may be referred to by more than one subroutine. Local display items are created, used, and cleared in the same routine. An example of a local display item is the tolerance circle used to test the fit of an opening.

The first five display items may be referenced from any portion of ALEXTR. These are:

Tag No.	Description
1	"END" light button.
2	"ACCEPT" light button.
3	A circle centered at 455, 512 with diameter = 305 (dimensions in rasters). This circle may represent the circumscribing circle of a section or the press container diameter.
4	A "+" located at the center of the above circle.
5	The extrusion section, scaled so as to just fit the above circle (Tag 3).

When the opening of a single opening die is being located or manipulated, the following tags are used:

Tag No.	Description
6	The opening scaled to the container diameter.
7	An "X" which marks the center of gravity of the opening.

APPENDIX IX

UTILITY PROGRAMS

Program NCDATA

NCDATA displays an NC data file on the CRT. The data displayed is the same as is used by the BCL NC milling machine to produce the electrode, die, or template. By displaying the cutter paths on the CRT, gross errors ay be checked for. It should be noted that NCDATA displays the center-line paths of the cutter specified for the part, and does not display the true part surface.

A feature of NCDATA is its ability to show the cutter paths in true isometric projection from any position relative to the first coordinate. This allows the viewer to "walk around" the surface and view it from any position. The view may also be scaled either up or down to permit the displayed image to appear as large as possible. The viewing position and scale must be specified by the user before the image is displayed. After the image is generated, the user may modify these parameters in order to view the surface from another position or at another scale.

Once the image is displayed, the user may translate the image both norizontally and vertically on the CRT. This is done using the light pen. Often, it is difficult to judge beforehand how an image should be positioned so that the entire image appears on the screen. Translation via the light pen allows the image to be centered on the screen after it is generated, in case the starting coordinates place part of the image off of the screen.

The following summarizes the inputs required from the user. The numerical responses (underlined values) are the values entered to generate the surface shown in Figure IX-1.

(1) File Name: This is the name of the NC output file generated by the EXTCAM (subroutine).

(2) Absolute or Incremental Data Format: All data generated by EXTCAM is incremental.

(3) Scale Factor: The amount by which the data is to be enlarged or reduced in order to fill the screen without going off the edges (0.7).

(4) Viewing Position: The coordinate position of the viewer along the three axis, relative to the first point of the display (-1, -1, 1).

APPENDIX IX

UTILITY PROGRAMS

Two stand-alone programs were developed to assist in preparing input for ALEXTR, and evaluating the output from EXTLAM. These programs are briefly described below.

Program ENTRDT

ENTRDT allows the user to create a data file in the format required by ALEXTR. It asks the user to enter the various values needed for a data file. The questions asked are as follows:

(1) Name of the file to be created. This must be entered as required by standard DEC procedures for file names (i.e., EXTRUS.DAT).

(2) 72 alpha-numeric characters for the file header. This could be the customer name and number, or other text information which describes the overall file.

(3) The section number as an integer value. Four digits are allowed for the section number. If a negative section number is entered, ENTRDT closes the data file and terminates.

(4) 72 alpha-numeric characters which describe the particular extrusion being defined.

(5) The X, Y coordinates and the radius value for each point on the polygon. These may either be integer or up to three-place decimal values. The three values are entered on the same line with comma separators. The program continues to accept the coordinate data, three at a time, until a value greater than 9000 is entered for X. When this is found, a section terminator is created in the file and the program loops back and asks for the next section number (Item 3 above).

(5) Starting Position of the Display on the CRT in Screen (Raster) Units: This locates the first point of the image and all other points are relative to it (500,350).

When the image is fully displayed, and after the light pen translation operation is completed, the viewer is given the opportunity to copy the displayed image onto the X-Y recorder. This provides him a hard-copy of the image he is viewing. After the image is copied, or if this step is skipped, the user is given the opportunity to generate a new display from the same file, but at a different scale or viewing position. In order to view a different file, it is necessary to terminate the current execution of NCDATA, and to re-RUN it.

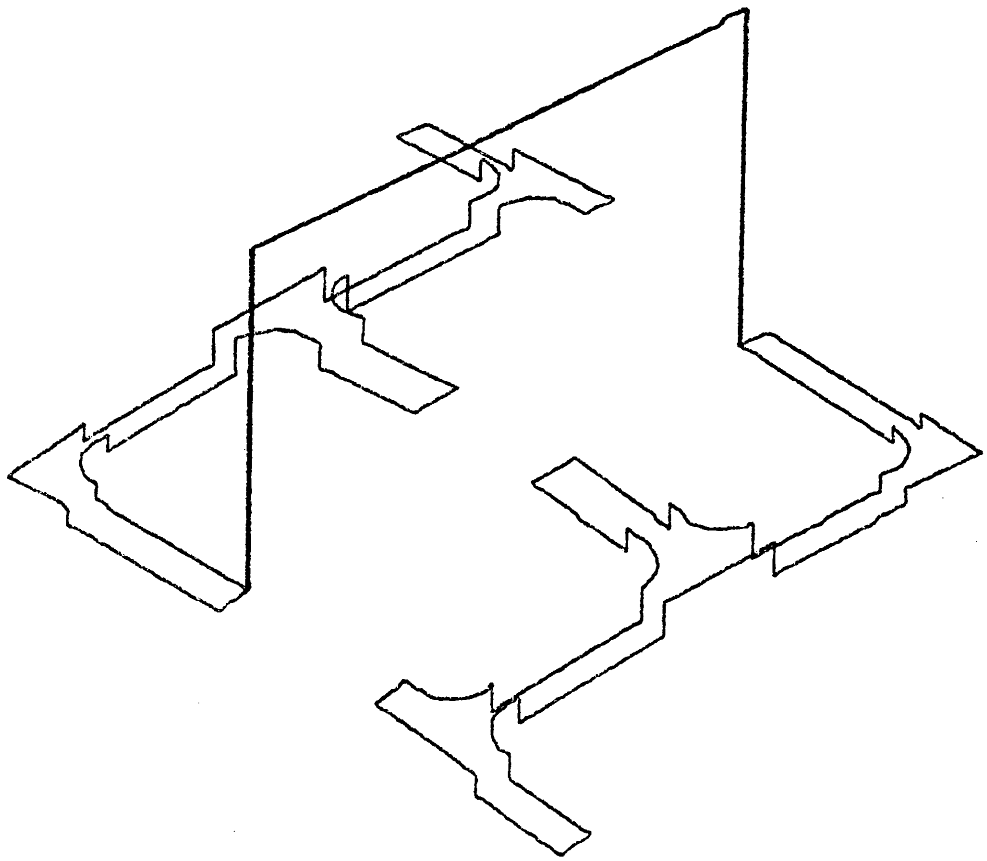


FIGURE IX-1. PLOT OF BEARING CUTTER PATHS GENERATED BY NCDATA

Army Materials & Mechanics Research Center
Watertown, Massachusetts 02172

AD____________________

UNCLASSIFIED
UNLIMITED DISTRIBUTION

COMPUTER-AIDED DESIGN AND MANUFACTURING FOR EXTRUSION OF ALUMINUM, TITANIUM, AND STEEL STURCTURAL PARTS
PHASE II APPLICATION TO PRACTICAL EXTRUSIONS

V. Nagpal, C. F. Billhardt and T. Altan
Battelle's Columbus Laboratories
Columbus, Ohio 43201

Technical Report - January 1978

102 pages, illustrations, tables, Contract DAAG46-76-0054
D/A Project 1497.94.5.S8154(XR5)

Final Report, July 19, 1976, to October 18, 1977

Key Words

Computer-Aided Design
CAD/CAM
Extrusion
Die Designs
Aluminum Alloys
Titanium Alloys
Steel

The overall objective of this program was to develop practical computer-aided design and manufacturing (CAD/CAM) techniques for extrusion of aluminum alloys, steels, and titanium alloys. This program was conducted in two phases. This report covers the Phase-II work, which was completed by performing the following major tasks: (a) assemble geometric modules to represent practical extrusion shapes, (b) apply the CAD/CAM method to a streamlined die, (c) develop a CAD/CAM system for non-lubricated extrusion through flat-face dies, (d) conduct extrusion trials and evaluate CAD/CAM extrusion results, and (e) evaluate the economics of CAD/CAM in extrusion. In order to enhance readability, the results of Phase-II work are presented in the form of two volumes. Volume I includes the following chapters: (1) CAD/CAM of Streamlined Dies for Lubricated Extrusion of "T" Sections, (2) CAD/CAM of Flat-Face Dies for Nonlubricated Extrusion of Aluminum Structural Shapes, and (3) Extrusion of "T" Sections of Aluminum, Titanium and Steel using computer-aided techniques. Volume II is the Instruction Manual and describes the content and use of computer programs.

Army Materials & Mechanics Research Center
Watertown, Massachusetts 02172

AD____________________

UNCLASSIFIED
UNLIMITED DISTRIBUTION

COMPUTER-AIDED DESIGN AND MANUFACTURING FOR EXTRUSION OF ALUMINUM, TITANIUM, AND STEEL STURCTURAL PARTS
PHASE II APPLICATION TO PRACTICAL EXTRUSIONS

V. Nagpal, C. F. Billhardt and T. Altan
Battelle's Columbus Laboratories
Columbus, Ohio 43201

Technical Report - January 1978

102 pages, illustrations, tables, Contract DAAG46-76-0054
D/A Project 1497.94.5.S8154(XR5)

Final Report, July 19, 1976, to October 18, 1977

Key Words

Computer-Aided Design
CAD/CAM
Extrusion
Die Designs
Aluminum Alloys
Titanium Alloys
Steel

The overall objective of this program was to develop practical computer-aided design and manufacturing (CAD/CAM) techniques for extrusion of aluminum alloys, steels, and titanium alloys. This program was conducted in two phases. This report covers the Phase-II work, which was completed by performing the following major tasks: (a) assemble geometric modules to represent practical extrusion shapes, (b) apply the CAD/CAM method to a streamlined die, (c) develop a CAD/CAM system for non-lubricated extrusion through flat-face dies, (d) conduct extrusion trials and evaluate CAD/CAM extrusion results, and (e) evaluate the economics of CAD/CAM in extrusion. In order to enhance readability, the results of Phase-II work are presented in the form of two volumes. Volume I includes the following chapters: (1) CAD/CAM of Streamlined Dies for Lubricated Extrusion of "T" Sections, (2) CAD/CAM of Flat-Face Dies for Nonlubricated Extrusion of Aluminum Structural Shapes, and (3) Extrusion of "T" Sections of Aluminum, Titanium and Steel using computer-aided techniques. Volume II is the Instruction Manual and describes the content and use of computer programs.

Army Materials & Mechanics Research Center
Watertown, Massachusetts 02172

AD____________________

UNCLASSIFIED
UNLIMITED DISTRIBUTION

COMPUTER-AIDED DESIGN AND MANUFACTURING FOR EXTRUSION OF ALUMINUM, TITANIUM, AND STEEL STURCTURAL PARTS
PHASE II APPLICATION TO PRACTICAL EXTRUSIONS

V. Nagpal, C. F. Billhardt and T. Altan
Battelle's Columbus Laboratories
Columbus, Ohio 43201

Technical Report - January 1978

102 pages, illustrations, tables, Contract DAAG46-76-0054
D/A Project 1497.94.5.S8154(XR5)

Final Report, July 19, 1976, to October 18, 1977

Key Words

Computer-Aided Design
CAD/CAM
Extrusion
Die Designs
Aluminum Alloys
Titanium Alloys
Steel

The overall objective of this program was to develop practical computer-aided design and manufacturing (CAD/CAM) techniques for extrusion of aluminum alloys, steels, and titanium alloys. This program was conducted in two phases. This report covers the Phase-II work, which was completed by performing the following major tasks: (a) assemble geometric modules to represent practical extrusion shapes, (b) apply the CAD/CAM method to a streamlined die, (c) develop a CAD/CAM system for non-lubricated extrusion through flat-face dies, (d) conduct extrusion trials and evaluate CAD/CAM extrusion results, and (e) evaluate the economics of CAD/CAM in extrusion. In order to enhance readability, the results of Phase-II work are presented in the form of two volumes. Volume I includes the following chapters: (1) CAD/CAM of Streamlined Dies for Lubricated Extrusion of "T" Sections, (2) CAD/CAM of Flat-Face Dies for Nonlubricated Extrusion of Aluminum Structural Shapes, and (3) Extrusion of "T" Sections of Aluminum, Titanium and Steel using computer-aided techniques. Volume II is the Instruction Manual and describes the content and use of computer programs.

Army Materials & Mechanics Research Center
Watertown, Massachusetts 02172

AD____________________

UNCLASSIFIED
UNLIMITED DISTRIBUTION

COMPUTER-AIDED DESIGN AND MANUFACTURING FOR EXTRUSION OF ALUMINUM, TITANIUM, AND STEEL STURCTURAL PARTS
PHASE II APPLICATION TO PRACTICAL EXTRUSIONS

V. Nagpal, C. F. Billhardt and T. Altan
Battelle's Columbus Laboratories
Columbus, Ohio 43201

Technical Report - January 1978

102 pages, illustrations, tables, Contract DAAG46-76-0054
D/A Project 1497.94.5.S8154(XR5)

Final Report, July 19, 1976, to October 18, 1977

Key Words

Computer-Aided Design
CAD/CAM
Extrusion
Die Designs
Aluminum Alloys
Titanium Alloys
Steel

The overall objective of this program was to develop practical computer-aided design and manufacturing (CAD/CAM) techniques for extrusion of aluminum alloys, steels, and titanium alloys. This program was conducted in two phases. This report covers the Phase-II work, which was completed by performing the following major tasks: (a) assemble geometric modules to represent practical extrusion shapes, (b) apply the CAD/CAM method to a streamlined die, (c) develop a CAD/CAM system for non-lubricated extrusion through flat-face dies, (d) conduct extrusion trials and evaluate CAD/CAM extrusion results, and (e) evaluate the economics of CAD/CAM in extrusion. In order to enhance readability, the results of Phase-II work are presented in the form of two volumes. Volume I includes the following chapters: (1) CAD/CAM of Streamlined Dies for Lubricated Extrusion of "T" Sections, (2) CAD/CAM of Flat-Face Dies for Nonlubricated Extrusion of Aluminum Structural Shapes, and (3) Extrusion of "T" Sections of Aluminum, Titanium and Steel using computer-aided techniques. Volume II is the Instruction Manual and describes the content and use of computer programs.

No. of Copies	To
	Project Manager, Navigation/Control Systems, Ft. Monmouth, New Jersey 07703
1	ATTN: DRCPM-NC-TM
	Commander, Avionics Research and Development Activity, Ft. Monmouth, New Jersey 07703
1	ATTN: DAVAA-O
	Director, Propulsion Laboratory, Research and Technology Laboratories (AVRADCOM), Mail Stop 77-5, 21000 Brookpark Rd., Cleveland, Ohio 44135
1	ATTN: DAVDL-LE
	Director, U.S. Army Industrial Base Engineering Activity, Rock Island Arsenal, Rock Island, Illinois 61201
1	ATTN: DRXIB-MT
	Air Force Materials Laboratory, Manufacturing Technology Division, Wright-Patterson Air Force Base, Ohio 45433
1	ATTN: AFML/LTM
1	AFML/LTN
1	AFML/LTE
	Commander, U.S. Army Electronics Command, Ft. Monmouth, New Jersey 07703
1	ATTN: DRSEL-RD-P
	Commander, U.S. Army Missile Command, Redstone Arsenal, Alabama 35809
1	ATTN: DRSMI-IIE
	Commander, U.S. Army Troop Support and Aviation Materiel Readiness Command, 4300 Goodfellow Blvd., St. Louis, Missouri 63120
1	ATTN: DRSTS-PLC
1	DRSTS-ME
2	DRSTS-DIL
	Commander, U.S. Army Armament Command, Rock Island, Illinois 61201
1	ATTN: DRSAR-PPP-1W
	Commander, U.S. Army Tank-Automotive Command, Warren, Michigan 48090
1	ATTN: DRSTA-RCM.1
12	Commander, Defense Documentation Center, Cameron Station, Building 5, 5010 Duke Street, Alexandria, Virginia 22314
	Hughes Helicopter, Division of Summa Corporation, Centinella Avenue and Teale Street, Culver City, California 90230
2	ATTN: Mr. R. E. Moore, Bldg. 314, M/S T-419
	Sikorsky Aircraft Division, United Aircraft Corporation, Stratford, Connecticut 06497
2	ATTN: Mr. Stan Silverstein, Section Supv, Manufacturing Tech.

DISTRIBUTION LIST
US ARMY AVIATION RESEARCH AND DEVELOPMENT COMMAND
MANUFACTURING TECHNOLOGY (MANTECH) PROJECTS
FINAL OR INTERIM REPORTS

No. of Copies	To
	Commander, U.S. Army Aviation Research and Development Command, P. O. Box 209, St. Louis, Missouri 63166
10	ATTN: DRDAV-EXT
1	DRDAV-EQ
	Project Manager, Advanced Attack Helicopter, P. O. Box 209, St. Louis, Missouri 63166
1	ATTN: DRCPM-AAH-TM
	Project Manager, Black Hawk, P. O. Box 209, St. Louis, Missouri 63166
1	ATTN: DRCPM-BH-T
	Project Manager, CH-47 Modernization, P. O. Box 209, St. Louis, Missouri 63166
1	ATTN: DRCPM-CH47M-T
	Project Manager, Aircraft Survivability Equipment, P. O. Box 209, St. Louis, Missouri 63166
1	ATTN: DRCPM-ASE-TM
	Project Manager, Cobra, P. O. Box 209, St. Louis, Missouri 63166
1	ATTN: DRCPM-CO-T
	Project Manager, Iranian Aircraft Program, P. O. Box 209, St. Louis, Missouri 63166
1	ATTN: DRCPM-IAP-T
	Commander, U.S. Army Materiel Development and Readiness Command, 5001 Eisenhower Avenue, Alexandria, Virginia 22333
4	ATTN: DRCMT
	Director, Applied Technology Laboratory, Research and Technology Laboratories (AVRADCOM), Ft. Eustis, Virginia 23604
1	ATTN: DAVDL-EU-TAS
	Director, Aeromechanics Laboratory, Research and Technology Laboratories (AVRADCOM), Mail Stop 215-1, Moffett Field, California 94035
1	ATTN: DAVDL-AM
	Director, Structures Laboratory, Research and Technology Laboratories (AVRADCOM), Mail Stop 266, Hampton, Virginia 23365

No. of Copies	To
	Director, Army Materials and Mechanics Research Center, Watertown, Massachusetts 02172
2	ATTN: DRXMR-PL
1	DRXMR-PR
1	DRXMR-PT
1	DRXMR-X
1	DRXMR-P
2	DRXMR-M
1	DRXMR-L
12	DRXMR-ER
2	DRXMR-TE
3	Authors

No. of Copies	To
	Bell Helicopter Textron, Division of Textron, Inc., P. O. Box 482, Ft. Worth, Texas 76101
2	ATTN: Mr. P. Baumgartner, Chief, Manufacturing Technology
	Kaman Aerospace Corporation, Bloomfield, Connecticut 06002
2	ATTN: Mr. A. S. Falcone, Chief of Materials Engineering
	Boeing Vertol Company, P. O. Box 16858, Philadelphia, Pennsylvania 19142
2	ATTN: R. Pinckney, Manufacturing Technology
	Detroit Diesel Allison Division, General Motors Corporation, P. O. Box 894, Indianapolis, Indiana 46206
2	ATTN: James E. Knott, General Manager
	General Electric Company, 10449 St. Charles Rock Road, St. Ann, Missouri 63074
2	ATTN: Mr. H. Franzen
	AVCO-Lycoming Corporation, 550 South Main Street, Stratford, Connecticut 08497
2	ATTN: Mr. V. Strautman, Manager, Process Technology Laboratory
	United Technologies Corporation, Pratt & Whitney Aircraft Division, Manufacturing Research and Development, East Hartford, Connecticut 06108
2	ATTN: Mr. Ray Traynor
	Grumman Aerospace Corporation, Bethpage, New York 11714
2	ATTN: Richard Cyphers, Manager, Manufacturing Technology, Plant 2
2	Albert Grenci, Manufacturing Engineer, Dept. 231, Plant 2
	Boeing Vertol Company, P. O. Box 16858, Philadelphia, Pennsylvania 19142
2	ATTN: R. Drago, Advanced Drive Systems Technology
	Lockheed Missiles & Space Co., Inc., 1111 Lockheed Way, Sunnyvale, California 94088
2	ATTN: H. Dorfman, Research Specialist, Manufacturing Research
1	Dr. Benjamin I. Bachrach, Ford Motor Company, Mfg. Processes Laboratory, Detroit, Michigan 48239
1	R. J. Livak, Consolidated Aluminum Corporation, 11960 Westline Industrial Drive, P. O. Box 14448, St. Louis, Missouri 63178
1	Mr. John Helfinstine, Senior Scientist, Research Dept., Corning Glass Works, Corning, New York 14830
1	John W. Durscher, Southwire Company, Carrollton, Georgia 30117